

УДК 62.501.72

В. И. ПОТАПОВ

Омский государственный
технический университет

ДИФФЕРЕНЦИАЛЬНАЯ ИГРА МЕЖДУ ПОДВИЖНЫМИ И НЕПОДВИЖНЫМИ ОБЪЕКТАМИ

Дана постановка игровой задачи типа «нападение — защита» для двух игроков, располагающих подвижными и неподвижными объектами соответственно. Приводится детальный алгоритм решения задачи методом дискретизации.

Ключевые слова: игровая задача, игроки, подвижные объекты, неподвижные объекты, алгоритм.

Рассмотрим следующую игру двух лиц. Игрок 1 располагает L управляемыми объектами, находящимися в начальных точках \bar{r}_{k0} , $1 \leq k \leq L$. Игрок 2 — N единицами защиты, которые он может расставлять в заданной области Γ , причем $\bar{r}_{k0} \in \Gamma$. Игрок 1 старается поразить k -м управляемым объектом ($1 \leq k \leq L$) заданную точку $\bar{r}_{kf} \in \Gamma$ которую защищает игрок 2, а игрок 2 старается помешать этому. Обозначим через t_{kf} время полёта k -го объекта от точки \bar{r}_{k0} до \bar{r}_{kf} . Число t_{kf} зависит от траектории $\bar{r}_k = \bar{r}_k(t)$, выбираемой k -м объектом, причем $\bar{r}_k(0) = \bar{r}_{k0}$, $\bar{r}_k(t_{kf}) = \bar{r}_{kf}$.

На траектории $\bar{r}_k = \bar{r}_k(t)$ наложим очевидные ограничения:

$$t_{kf} \leq T_k, \quad (1)$$

$$|\dot{\bar{r}}_k(t)| \leq M_k \text{ для любого } t \in [0, t_{kf}]. \quad (2)$$

Неравенство (1) диктуется ресурсом двигателя, которым располагает k -й подвижный объект, управляемый игроком 1; неравенство (2) отражает прочностные характеристики k -го подвижного объекта.

Из (2) следует, что для любого $t \in [0, t_{kf}]$

$$|\dot{\bar{r}}_k(t)| \leq M_k t + v_{k0}, \quad (3)$$

$$|\bar{r}_k(t)| \leq M_k t^2 / 2 + v_{k0} t,$$

где v_{k0} — начальная скорость подвижного объекта.

Введем, наконец, последнее ограничение на траектории:

$$\bar{r}_k(t) \in V \text{ для любого } t \in [0, t_{kf}], \quad (4)$$

где V — заданная конечносвязная область в R^3 , причем $\bar{r}_{k0}(t) \in V$ и $\Gamma \subset V$. Это ограничение «запрещает»

для траектории некоторые односвязные области пространства.

Пусть, далее, множество пунктов защиты игрока 2 может быть расположено в Q точках ($N \leq Q$) с заданными координатами $\bar{c}_i = (c_{1i}, c_{2i}, c_{3i})$, $1 \leq i \leq Q$. Обозначим множество $\{\bar{c}_1, \bar{c}_2, \dots, \bar{c}_Q\}$ через C .

Итак, игроки 1 и 2 обладают каждый L А[1, $\lambda_{kq}(t, \bar{r}_k)$]-системами соответственно ($q = 1, 2$), поведение которых, при аппроксимации марковским процессом [1], описывается уравнениями

$$p'_{k1} = -\lambda_{k1}(t, \bar{r}_k)p_{k1}, \quad 1 \leq k \leq L,$$

$$p'_{k2} = -\lambda_{k2}(t, \bar{r}_k)p_{k2},$$

с начальными условиями $p_{k1}(0) = p_{k2}(0) = 1$, где $p_{kq}(t)$ — вероятность безотказной работы k -й А-системы, принадлежащей q -му игроку ($1 \leq k \leq L$; $q = 1, 2$) к моменту времени t , а $\lambda_{kq}(t, \bar{r}_k)$ — интенсивность отказов этой системы, где $\bar{r}_k = \bar{r}_k(t)$ — траектория k -го объекта, управляемого игроком 1.

Интенсивности отказов систем зададим следующим образом:

$$\lambda_{k1}(t, \bar{r}_k) = \lambda_k(t, \bar{r}_k) + \sum_{i=1}^N \frac{b_i(|\bar{r}_k - \bar{c}_i^*|)}{|\bar{r}_k - \bar{c}_i^*|^{\alpha(\bar{c}_i^*)}},$$

$$\lambda_{k2}(t, \bar{r}_k) = \frac{b_{k2}}{|\bar{r}_k - \bar{r}_{kf}|^{\beta_k}},$$

где $\lambda_k(t, \bar{r}_k)$ — интенсивность отказов k -го подвижного объекта, определяемая объективными факторами; $C^* = \{\bar{c}_1^*, \bar{c}_2^*, \dots, \bar{c}_N^*\}$ — некоторые подмножества множества C (пункты защиты игрока 2); $\{\alpha(\bar{c}_i^*)\}$ — заданная последовательность действительных чисел, отвечающая множеству C^* , элементы которой определяются физико-географическими или иными особенностями точки \bar{c}_i^* ; $\{b_{k2}\}$ и $\{\beta_k\}$ — заданные последовательности действительных чисел, элементы которых определяются физико-географическими или иными особенностями точек \bar{r}_{kf} , подлежащих защите, и типом k -го подвижного объекта;

$$b_i(x) = \begin{cases} 0, & \text{если } x > \gamma_i, \\ 1, & \text{если } 0 \leq x \leq \gamma_i, \end{cases}$$

где γ_i — дальность действия пункта защиты \bar{c}_i^* игрока 2.

Интенсивность отказов $\lambda_{k1}(t, \bar{r}_k)$ можно представить в виде

$$\lambda_{k1}(t, \bar{r}_k) = \lambda_k(t, \bar{r}_k) + \sum_{i=1}^Q \frac{\delta_i b_i(|\bar{r}_k - \bar{c}_i^*|)}{|\bar{r}_k - \bar{c}_i^*|^{\alpha_i}},$$

где $\bar{\delta} = (\delta_1, \delta_2, \dots, \delta_Q)$ — вектор, определяемый следующим образом:

$$\delta_i = \begin{cases} 0, & \text{если в точке } \bar{c}_i \text{ нет пункта защиты,} \\ 1 & \text{в противном случае,} \end{cases}$$

а $\alpha_i = \alpha(\bar{c}_i)$.

В качестве функции выигрыша возьмем

$$K[z_1, z_2] = \sum_{k=1}^L p_{k1}(t_{kf}) - \sum_{k=1}^L p_{k2}(t_{kf}),$$

где $z_q \in W^q$, а W^q — множества стратегий q -го игрока ($q = 1, 2$).

Ясно, что

$$W^1 = \{\bar{r}_1(t), \bar{r}_2(t), \dots, \bar{r}_L(t)\},$$

$$W^2 = \{\bar{\delta}_1, \bar{\delta}_2, \dots, \bar{\delta}_R\},$$

где $R = \binom{Q}{N}$.

Далее, обозначим $v_{k0} = |\bar{r}_k(0)|$ и ε — точность измерения траектории $\bar{r}_k = \bar{r}_k(t)$, при которой становится заметно отклонение траектории от касательной в окрестности произвольной точки $t_0 \in [0, t_{kf}]$.

Теперь для решения задачи можно применить аппарат дискретизации, разработанный в [2] для решения подобных задач.

За временной интервал дискретизации возьмем число $\Delta t = \sqrt{2\varepsilon/M}$, где

$$M = \max\{M_1, M_2, \dots, M_L\}.$$

Получим оценки для t_{kf} .

Очевидно, что

$$|\bar{r}_k(t_f)| = |\bar{r}_{kf}| \leq \frac{M_k t_{kf}^2}{2} + v_{k0} t_{kf}.$$

Отсюда следует

$$t_{kf} \geq \frac{1}{M_k} \left(-v_{k0} + \sqrt{v_{k0}^2 + 2M_k |\bar{r}_{kf}|} \right).$$

Пространственным аналогом временного интервала дискретизации будет шар с радиусом $\rho(\Delta t)$, где

$$\rho(\Delta t) = (v_0 + \tilde{M}\Delta t / 2)\Delta t,$$

$$v_0 = \min\{v_{10}, v_{20}, \dots, v_{L0}\},$$

$$\tilde{M} = \min\{M_1, M_2, \dots, M_L\}.$$

Решением задачи является вычисление стратегий \tilde{z}_1 и \tilde{z}_2 , для которых

$$K[\tilde{z}_1, \tilde{z}_2] \min_{z_2} \max_{z_1} K[z_1, z_2].$$

За основу приведенного ниже алгоритма для решения этой задачи, принят разработанный автором в [2] алгоритм 1.

АЛГОРИТМ

1. Задать $\{\lambda_1(t, \bar{r}); \lambda_2(t, \bar{r}), \dots, \{\bar{r}_{10}, \bar{r}_{20}, \dots, \bar{r}_{L0}\}; \{v_{10}, v_{20}, \dots, v_{L0}\}; \{\bar{r}_{10}, \bar{r}_{20}, \dots, \bar{r}_{L0}\}; \{M_1, M_2, \dots, M_L\}; \{T_1, T_2, \dots, T_L\}; \{\gamma_1, \gamma_2, \dots, \gamma_Q\}; \{b_{12}, b_{22}, \dots, b_{L2}\}; \{\beta_1, \beta_2, \dots, \beta_L\}; \{\bar{c}_1, \bar{c}_2, \dots, \bar{c}_Q\}; \{\alpha_1, \alpha_2, \dots, \alpha_Q\}; \varepsilon, N, V$.
2. Вычислить Δt ; $\rho(\Delta t)$ по приведенным выше формулам.
3. Вычислить

$$t_{kf}^{\min} = \frac{1}{M_k} \left(-v_{k0} + \sqrt{v_{k0}^2 + 2M_k |\bar{r}_{kf}|} \right), \quad 1 \leq k \leq L.$$

4. Вычислить

$$\tilde{r}_k = [(T_k - t_{kf}^{\min}) / \Delta t], \quad 1 \leq k \leq L.$$

5. Выполнить процедуру 6–24 этого алгоритма для всех векторов $\bar{\delta} = (\delta_1, \delta_2, \dots, \delta_Q)$, где $\delta_i = 0, 1$ и $\delta_1 + \delta_2 + \dots + \delta_Q = N$, $1 \leq i \leq Q$.

6. Положить $k = 1$.
7. Положить $j = 0$.
8. Вычислить $t_{kf} = t_{kf}^{\min} + j \Delta t$.
9. Положить $\bar{i}_0 = 0$.
10. Положить $v = 1$.
11. Положить $n = v$.
12. Вычислить $\lambda_{k1}^v, \lambda_{k2}^v$ по формулам

$$\lambda_{kq}^v = \left(\frac{\pi^2(\rho^2 + (\Delta t)^2)^2}{2} \right)^{-1} \int_{t_{v-1}}^{t_v} dt \int_{|\bar{r}-\bar{i}| \leq \rho} \lambda_{kq}(t, \bar{r}) dv,$$

где $t_v = v \Delta t$; $t_0 = 0$; $\bar{r}_k = \varepsilon \bar{i}_{n-1}$; $\bar{r} = (x_1, x_2, x_3)$;
 $dv = dx_1, dx_2, dx_3$; $q = 1, 2$.

13. Для всех целочисленных векторов $\bar{i} = (i_1, i_2, i_3) \in I_n$, где $I_n = I_n^1 \cap I_n^2 \cap \bar{V}$, а множества I_n^1, I_n^2, \bar{V} определяются следующим образом:

$$I_n^1 = \left\{ \bar{i} \mid \|\bar{i} - \bar{i}_{n-1}\| \leq \frac{\rho}{\varepsilon} \right\};$$

$$I_n^2 = \left\{ \bar{i} \mid \|\bar{i}_{kf} - \bar{i}\| \leq \frac{1}{\varepsilon} (M_k t_{kf} + v_{k0})(t_{kf} - t_v) \right\};$$

$$\bar{V} = \{ \bar{i} \mid \varepsilon \bar{i} \in V \}; \bar{r}_{kf} = \varepsilon \bar{i}_{kf},$$

выполнить процедуру 14–15 этого алгоритма.

14. Если $I_n = \emptyset$, идти к 19.
15. Вычислить

$$K_j[\bar{r}_k(t_v)] = \rho_{k1}^v(t_v) - \rho_{k2}^v(t_v),$$

где $\rho_{kq}^v(t_v) = \rho_{kq}^v(t_{v-1}) \exp(-\lambda_{kq}^v \Delta t)$, $q = 1, 2$;

$$\rho_{kq}^v(t_{v-1}) = \rho_{kq}^{v-1}(t_{v-1});$$

$$\rho_{kq}^0(0) = 1, \bar{r}(t_v) = \varepsilon \bar{i}.$$

16. Вычислить вектор $\bar{i}_n \in I_n$, для которого

$$K_j[\bar{r}_{kn}(t_v)] = \max_{i \in I_n} K_j[\bar{r}_k(t_v)],$$

$$\bar{r}_{kn}(t_v) = \varepsilon \bar{i}_n.$$

17. Положить $v = v + 1$.
18. Если $t_v \leq t_{kf}$, идти к 11.

19. Положить $j = j + 1$.
20. Если $j \leq \bar{1}_k$, идти к 8.
21. Вычислить \bar{j} , для которого

$$K_j[\bar{r}_k(t_{kf})] = \max_{0 \leq j \leq \bar{1}_k} K_j[\bar{r}_k(t_{kf})].$$

22. Положить $k = k + 1$.
23. Если $k \leq L$, идти к 7.
24. Вычислить

$$G[\bar{\delta}] = \sum_{k=1}^L K_j[\bar{r}_k(t_{kf})].$$

25. Вычислить $\bar{\delta}_0$, для которого

$$G[\bar{\delta}] = \min_{\bar{\delta}} G[\bar{\delta}].$$

26. Конец $\bar{\delta}_0$ и $\{\bar{r}_1(t), \bar{r}_2(t), \dots, \bar{r}_L(t)\}$ — искомые оптимальные стратегии.

В пояснение к алгоритму заметим, что $\rho_{kq}^v(t)$ в п. 15 есть решение уравнений

$$(\rho_{kq}^v)' = -\lambda_{kq}^v \rho_{kq}^v; q = 1, 2,$$

с начальными условиями $\rho_{kq}^v(t_{v-1}) = \rho_{kq}^{v-1}(t_{v-1})$, которые являются дискретными аналогами уравнений (5), описывающих игру, на интервалах дискретизации.

Библиографический список

1. Вентцель. Е. С. Исследование операций / Е. С. Вентцель. — М. : Сов. радио, 1972. — 550 с.
2. Потапов, В. И. Дифференцированная игра между управляемыми подвижными объектами / В. И. Потапов, С. Г. Братцев; Омский политехнический институт. — Омск, 1985. — 31 с. — Деп. в ВИНТИ 17.0785, № 6002-85.

ПОТАПОВ Виктор Ильич, доктор технических наук, профессор (Россия), заслуженный деятель науки и техники РФ, заведующий кафедрой «Информатика и вычислительная техника».

Адрес для переписки: ivt@omqtu.ru

Статья поступила в редакцию 11.05.2012 г.

© В. И. Потапов

Книжная полка

Губарев, В. В. Информатика: прошлое, настоящее, будущее : учеб. для вузов по направлению 230100 «Информатика и вычислительная техника» и специальности 080801 «Прикладная информатика» и др. экон. специальностям / В. В. Губарев. — М. : Техносфера, 2011. — 431 с. — ISBN 978-5-94836-288-5.

В пособии излагается взгляд автора на то, что такое информатика, ее состав, основные понятия, концепция описания ее истории и поколений средств вычислительной техники. Особое внимание уделяется хронологии создания базовых средств и технологий информатики, сведениям о лицах, внесших весомый вклад в развитие разных разделов информатики, а также ближайшим перспективам их развития. Содержатся многочисленные справочные, в частности статистические, сведения и перечень междисциплинарных проблемных вопросов, касающихся понятия информации, информатики и ее разделов. В книгу включены учебные и справочные материалы, предназначенные для изучения в рамках учебного процесса при подготовке бакалавров и магистров по направлениям «Информатика и вычислительная техника», «Прикладная математика и информатика», «Программная инженерия», «Системный анализ и управление», «Информационные системы и технологии», «Прикладная информатика» (по отраслям), «Математическое обеспечение и администрирование информационных систем», «Управление в технических системах», «Бизнес-информатика», «Информационная безопасность» и т.д.

РАЗРАБОТКА ЭКСПЕРТНОЙ СИСТЕМЫ ДЛЯ ОЦЕНКИ КАЧЕСТВА ПОСТЕЛЬНОГО БЕЛЬЯ

В статье рассмотрены вопросы разработки экспертной системы для оценки качества готовых швейных изделий. Представлены структура, схема действия экспертной системы на примере постельного белья. Отражены новизна результатов исследования, их практическая, социальная и экономическая значимость.

Ключевые слова: экспертная система, база знания, база данных, механизм логического вывода, контроль качества, швейные изделия.

В связи с переходом российских предприятий к международным стандартам ИСО серии 9000: 2000 и возрастающими требованиями к качеству готовой продукции, используемая на швейных предприятиях современная концепция оценки качества не всегда является достаточно эффективной. В связи с этим цель данного исследования заключается в необходимости уделять большее внимание аттестации готовой продукции.

Анализ современных достижений в области контроля качества швейных изделий показал, что исследование ведутся в различных направлениях, начиная с этапа предпроектных исследований и до реализации конечной продукции.

Наиболее современным и малоизученным направлением совершенствования деятельности швейных предприятий является применение экспертных систем (ЭС) в аттестации готовой продукции [1].

В настоящее время разработана архитектура ЭС для оценки качества чертежей конструкций одежды [2]. Создана ЭС, которая позволяет осуществить автоматизацию поиска решений и синтеза оригинальных моделей с использованием информации, хранящейся в базах знания в виде качественных признаков [3]. Представлено методологическое и информационное обеспечение ЭС для выбора предпочтительных моделей в зависимости от признаков внешнего облика и антропоморфологических особенностей фигур потребителей; разработана система контроля качества визуального образа модели (в системе «человек – одежда») для прогнозирования показателей качества с учетом внешнего облика заказчика на стадии эскизного проектирования [4].

В данной статье описаны требования к структуре и формированию базы знания (БЗ) ЭС для аттестации постельного белья. В целях решения задачи разработки ЭС были изучены и проанализированы стандартизованные и экспертные данные, необходимые для контроля качества постельного белья, а также источники по теории разработки ЭС.

Цель создания ЭС состоит в возможности значительного усиления интеллектуального потенциала человека для формулирования необходимых заключений, при наличии не формализуемых параметров и ограничений, характерных для мышления человека, а также расширение компетенции эксперта-чело-

века, не имеющего большого опыта в области оценки качества постельного белья.

Традиционно ЭС состоит из двух подсистем: базы знаний (БЗ) и механизма логического вывода (МЛВ). МЛВ предназначен для принятия решения на основе опытных знаний и сведений, уникальных и специфических для решаемой проблемы [5].

Первым шагом к созданию ЭС является разработка первоначального прототипа, представленного с помощью информации, сформированной на основе знаний опытного эксперта и анализа стандартизованных знаний, а затем — последовательное совершенствование модели на основе обратной связи как с экспертом, так и с конечными пользователями ЭС.

На базе этих данных можно разработать ЭС, с помощью которой удастся систематизировать технологический процесс аттестации качества различных видов одежды.

Ассортимент швейных изделий чрезвычайно многообразен, включает различные виды одежды, головные уборы, а также постельное, столовое белье и пр. Швейные изделия подразделяют на классы, подклассы, группы, подгруппы и др. Последней ступенью классификации является артикул (номер изделия) [6]. Все разнообразие швейных изделий необходимо подвергать оценке качества, а этот процесс трудно формализуем в связи с постоянным, в том числе сезонным, обновлением ассортимента.

В каждой группе швейных изделий используются специфические критерии оценки, следовательно, с целью систематизации процесса контроля качества следует начать с наиболее простого, с точки зрения формализации, ассортимента, а затем продолжить работу с более сложными изделиями. Именно таким требованиям соответствует постельное белье, на примере которого проще структурировать информацию о дефектах, сформулировать правила, и, по мере усложнения ассортимента, наращивать БЗ, в которой, кроме информации общего характера, должно быть представлено описание всех возможных дефектов и рекомендации по их устранению, т. е. создано соответствующее информационное обеспечение.

Современный контроль качества промышленной продукции традиционно сводится к оценке соответствия выполнения технологических операций ГОСТ и ТУ, в которых основными показателями качества



Рис. 1. Последовательность работы ЭС

изготовления являются: дефекты внешнего вида, производственно-швейные и пороки материалов. К ним относятся: кривизна строчек, пропуски стежков в швах, текстильные пороки, перекосы, несимметричность парных деталей и т.д. [1].

Практическим результатом анализа является разработка исследовательского прототипа ЭС, которая состоит из 12 баз данных, содержащих информацию о типах, размерах, допустимых отклонениях, особенностях технологической обработки по ГОСТ 31307-2005 [7], а также графическую информацию о используемых методах обработки, сведения о дефектах и способах их устранения [6].

Работа ЭС осуществляется в диалоговом режиме. После запуска программы появляется окно «Постельное белье», в котором можно выбрать два варианта диалога. Выходной информацией, в первом варианте, является таблица с результатами поэтапного анализа тестируемых предметов постельного белья. Во втором варианте, после того как эксперт рассмотрит все интересующие вкладки, формируется характеристика тестируемого объекта. По результатам анализа можно сделать вывод о пригодности изделий для реализации потребителю.

Последовательность работы ЭС представлена схемой (рис. 1) [8, 9].

Описанная ЭС содержит режим наполнения графическими изображениями и текстовой информацией без участия программиста, БЗ экспертной системы включает только сведения из стандартов и знания эксперта, но на предприятиях существуют свои требования, характерные для того или иного способа производства. Поэтому эксперт может дополнить БЗ опытными данными из технических условий и (или) технических описаний конкретного предприятия.

Преимущества разработанной ЭС является систематизация предметов постельного белья, методов обработки, описание и графические изображения которых позволят легко распознать дефекты, если они выявлены.

Человек, который выступает в данном случае экспертом, может использовать рассматриваемую программу как учебное пособие по оценке качества постельного белья. МЛВ выполняет функции поиска в БЗ, операции над знаниями и получение заключений; обеспечивает формирование выводов, воспринимая вводимые факты как элементы правил, отыс-

кивая правила, в состав которых входят введенные факты, и, актуализируя те части БЗ, которым соответствуют введенные факты [5].

Задача эксперта при использовании системы — правильно формулировать условия и извлекать необходимые данные. ЭС применяются для решения только трудных практических задач, по качеству и эффективности результаты ЭС не уступают выводам эксперта-человека. Выводы ЭС обладают «прозрачностью», могут быть объяснены пользователю на качественном уровне. Это свойство обеспечивается их способностью демонстрировать процесс принятия решений [6].

Экономическая и социальная значимость разработанной ЭС состоит в возможности расширения квалификации эксперта, а также в повышении удовлетворенности населения качеством швейных изделий.

Новизна данного исследования заключается в формализации процесса оценки качества постельного белья на этапе готовой продукции, алгоритмизация и автоматизация процесса контроля для указанного ассортимента, а также формирование структуры базы знаний ЭС.

В настоящее время аттестация качества на швейных предприятиях города Омска осуществляется не на всех стадиях производства и, чаще всего, сводится к органолепическим методам, которых, в большинстве случаев, недостаточно для создания конкурентоспособной продукции. Из-за отсутствия квалифицированных специалистов в данной области, а также нежелания вложения дополнительных денежных средств и затрат времени на многих предприятиях выходной контроль качества не осуществляется вообще.

Практическая значимость исследования заключается в разработке информационного и программного обеспечения для использования на швейных предприятиях.

Описанная ЭС позволит решить проблему оценки качества швейных изделий в соответствии с требованиями, предъявляемыми к конкретному ассортименту.

Библиографический список

1. Астапенко, Н. А. Перспективы развития системы оценки качества швейных изделий с использованием экспертных систем : сб. / Н. А. Астапенко // Молодежь, наука, творчество — 2011 : сб. материалов. Ч. 1. — Омск : Омский гос. ин-т сервиса, 2011. — С. 22–24.

2. Гниденко, А. В. Формализация и алгоритмизация процесса проектирования женской одежды с рукавами сложных покровов : автореф. дис. ... канд. тех. наук / А. В. Гниденко. — Иваново : Ивановская гос. текстильн. академия, 2008. — 294 с.

3. Нигматова, Ф. У. Формирование промышленного ассортимента одежды на базе экспертной системы / Ф. У. Нигматова, Х. А. Алимова // Швейная промышленность. — М. : Легпроминфо, 2009. — № 2. — С. 56–58.

4. Кривобородова, Е. Ю. Разработка методологии адресного проектирования одежды с использованием новых информационных технологий : автореф. дис. ... канд. тех. наук / Е. Ю. Кривобородова. — М. : Московский гос. ун-т дизайна и технологии, 2004. — 362 с.

5. Андрейчиков, А. В. Интеллектуальные информационные системы : учеб. / А. В. Андрейчиков, О. Н. Андрейчиков. — М. : Финансы и статистика, 2004. — 424 с.

6. Силаева, М. А. Пошив изделий по индивидуальным заказам / М. А. Силаев // Технология одежды : учеб. для начального профессионального образования. — М. : Академия, 2006. — 528 с.

7. ГОСТ 31307-2005. Белье постельное. Общие технические условия. — Введен впервые 2005.01.01. — М. : Стандартиформ, 2005. — 16 с.

8. ГОСТ 19.701-90. Схемы алгоритмов, программ, данных и систем. Условные обозначения и правила выполнения. — Взамен ГОСТ 19.002-80, ГОСТ 19.003-80 ; введ. 1992–01–01. — М. : Госстандарт СССР, 1992. — 26 с.

9. Словарь IT терминологии: блок-схема [Электронный ресурс]. — Режим доступа: <http://it2009.ru/view.php?id=350>. (дата обращения: 11.09.2010).

ГОНЧАРОВА Надежда Алексеевна, аспирантка, старший лаборант кафедры конструирования швейных изделий.

РЕВЯКИНА Ольга Владимировна, кандидат технических наук, доцент (Россия), декан заочного факультета.

Адрес для переписки: jem4ug87@mail.ru;

Статья поступила в редакцию 08.12.2011 г.

© Н. А. Гончарова, О. В. Ревякина

Статья поступила в редакцию 08.12.2011 г.

© Н. А. Гончарова, О. В. Ревякина

Книжная полка

Гаврилов, М. В. Информатика и информационные технологии : учеб. для бакалавров вузов по юрид. специальностям / М. В. Гаврилов, В. А. Климов. — 2-е изд., испр. и доп. — М. : Юрайт, 2012. — 350 с. — ISBN 978-5-9916-1559-4.

Учебник подготовлен в рамках новых государственных образовательных стандартов по специальностям высшего и среднего профессионального образования. Изложены базовые понятия по информатике, информационным технологиям, современным компьютерным аппаратным средствам. Раскрыты назначение, возможности применения и дана классификация программного обеспечения, рассмотрены операционная система Microsoft Windows, прикладные программы различного назначения последних версий. Строгая формулировка основных понятий сочетается с доходчивыми пояснениями и рекомендациями по практической работе. Подробно изложены вопросы организации размещения, обработки, хранения и передачи информации. Описаны услуги глобальных компьютерных сетей, сети Интернет. Особое внимание уделено законодательной и технической защите от несанкционированного доступа, средствам антивирусной защиты.

ДВУХШАГОВЫЙ ЭКСТРАГРАДИЕНТНЫЙ МЕТОД С ПАМЯТЬЮ ДЛЯ РЕШЕНИЯ ВАРИАЦИОННЫХ НЕРАВЕНСТВ СО СВЯЗАННЫМИ ОГРАНИЧЕНИЯМИ

В работе предлагается и обосновывается двухшаговый экстраградиентный метод с памятью для решения вариационных неравенств со связанными ограничениями. Предложенный метод допускает эффективную численную реализацию на многопроцессорных вычислителях. В качестве практического примера использования вариационных неравенств со связанными ограничениями рассмотрена несобственная задача математического программирования. Эта задача реализует модель планирования производства, в которой внешняя рыночная стоимость ресурсов совпадает с внутренними объективно обусловленными оценками ресурсов.

Ключевые слова: вариационное неравенство, экстраградиентный метод, оптимизация, сходимость, несобственная задача математического программирования.

Работа выполнена при финансовой поддержке РФФИ, проект 12-07-00326.

Введение. Итерационные методы являются универсальным подходом к решению вариационных неравенств. Одна итерация метода заключается в вычислении направления в заданной итерационной точке и величины шага вдоль вычисленного направления для того, чтобы совершить переход в очередную итерационную точку. Собственно, итерационные методы различаются в способе выбора величины шага (постоянный, переменный, результат решения вспомогательной задачи) или направления (градиентные, сопряженных градиентов, экстраградиентные методы).

Применение параллельных вычислений в классических градиентных и экстраградиентных методах может быть эффективно только при распараллеливании трудоемких операций на текущей итерации и неэффективно на нескольких итерациях одновременно, так как вычисление очередной итерационной точки возможно только после вычисления предыдущей. Классические градиентные и экстраградиентные методы никак не используют информацию о направлении, в котором они двигались на предыдущей итерации и каждый раз вычисляют направление заново. Если же запоминать вычисленное ранее направление, то можно построить вычислительную схему, пригодную для использования параллельных вычислений. В статье для решения вариационных неравенств со связанными ограничениями предлагается эффективная для параллельной реализации вычислительная схема двухшагового экстраградиентного метода с памятью.

Постановка задачи. Пусть итерационный процесс имеет вид

$$x_{k+1} = \Phi(x_k),$$

где Φ — некоторая процедура, последовательность действий, необходимая для получения следующей итерационной точки.

Вектором движения итерационного процесса на k итерации назовем разность точек, полученных на k и $k-1$ итерациях:

$$r_k = x_k - x_{k-1}.$$

Используя введенное определение, построим на k итерации вспомогательную точку $x'_k = x_k + r_k$. Тогда очередную итерационную точку x_{k+1} вычислим следующим образом:

$$x_{k+1} = \Phi(x'_k), \text{ если } \|\Phi(x_k) - x'_k\| \leq \sigma,$$

$$x_{k+1} = \Phi(x_k), \text{ если } \|\Phi(x_k) - x'_k\| > \sigma,$$

где $\sigma > 0$. Выполнение условия $\|\Phi(x_k) - x'_k\| \leq \sigma$ будем называть критерием одного направления.

Идея предлагаемого подхода заключается в том, что, используя вычисленное ранее направление, можно легко вычислить вспомогательную точку. Затем вычисляем $\Phi(x_k)$, а на другом процессоре — $\Phi(x'_k)$. И если $\Phi(x_k)$ и x'_k оказываются достаточно близки, то в качестве следующей точки следует брать $\Phi(x'_k)$. Данный подход позволяет существенно сократить число итераций в тех задачах, в которых вектор движения итерационного процесса меняется незначительно. Также возможно его применение совместно с распараллеливанием трудоемких операций на каждой итерации. Методы с применением предлагаемого подхода назовем методами с памятью.

Несобственная задача математического программирования. Связанные ограничения возникают для задач, у которых наряду с внутренними параметрами имеются внешние параметры, выражающие влияние управляющих воздействий внешней среды на процесс принятия решений. Решение этой проблемы приводит к различным равновесным конструкциям, таким, как задачи о вычислении неподвижных точек, обратные задачи оптимизации и др. [1].

При классическом подходе решение задачи определяется для каждого фиксированного набора внешних параметров. В этом случае затруднено определение равновесия между внутренними объективно обусловленными оценками и внешними управляющими воздействиями. Эффективное решение этой проблемы — использование вариационных неравенств со связанными ограничениями как наиболее общей формы записи для таких задач. Структура полученных математических конструкций принципиально отличается от классических конструкций двойственности для задач оптимизации, поэтому эффективное использование таких задач требует разработки новых методов для нахождения решений этих задач.

Рассмотрим задачу выпуклого программирования (ВП)

$$\min \{f_0(x) | f_1(x) \leq b_1, \dots, f_m(x) \leq b_m, x \in X\}, \quad (1)$$

где выпуклое множество $X \subseteq R^n$, $f_0, f_1, \dots, f_m : X \rightarrow R$ — выпуклые функции.

Общепринятая экономическая интерпретация задачи ВП — это модель производства, в которой требуется выбрать интенсивности $x = (x_1, \dots, x_n)$ работы предприятия такие, чтобы обеспечить выпуск продукции в соответствии с технологическим процессом $F(x) = (f_1(x), \dots, f_m(x))$ из ограниченных запасов ресурсов $b = (b_1, \dots, b_m)$ с минимальными издержками $f_0(x)$.

Задача (1) широко известна, и существует множество методов ее решения. Однако в случае противоречивости исходной модели (например, когда количество имеющихся ресурсов не позволяет выполнить выбранные технологические процессы) система ограничений будет несовместной, а соответствующая задача оптимизации (1) становится несобственной [2]. Одним из подходов, предложенных академиком И. И. Ерёминым для коррекции таких задач, является параметризация исходной задачи и определение параметров, обеспечивающих разрешимость задачи. При этом можно дополнительно оптимизировать получаемую в результате коррекцию задачи.

Для несобственной задачи математического программирования (1) введем параметризацию через параметры $y = (y_1, \dots, y_m)$, характеризующие рыночную стоимость ресурсов, тогда объемы запасов ресурсов b естественно рассматривать как функцию от параметров y . Кроме того, издержки производства свяжем по координатам с ресурсами, что тоже вполне естественно, так как с каждым видом ресурса могут быть связаны свои производственные затраты. Итак, пусть $F_0(x) = (f_{01}(x), \dots, f_{0m}(x))$, $F_1(x) = (f_{11}(x), \dots, f_{1m}(x))$, $P(x) = (p_1(x), \dots, p_m(x))$, $x \in X$, $X \subseteq R^n$, X — выпуклое множество, $y \in R^m_+$, $f_{0i}(x)$ и $f_{1i}(x)$, $i=1, \dots, m$, — выпуклые скалярные функции. Функции $f_{0i}(x)$, $i=1, \dots, m$ задают издержки производства на единицу рыночной стоимости y_i для каждого вида ресурса, при возрастании стоимости y_i увеличиваются и издержки $y_i f_{0i}(x)$, сумма которых задает суммарные издержки. Функции $f_{1i}(x)$, $i=1, \dots, m$ определяют технологический процесс или расход ресурсов на производство с интенсивностями $x = (x_1, \dots, x_n)$, функции $p_i(y)$, $i=1, \dots, m$ — запасы ресурсов в зависимости от их рыночной стоимости $y = (y_1, \dots, y_m)$.

В результате получаем задачу параметрического выпуклого программирования

$$\min \{y^T F_0(x) | F_1(x) \leq P(y), x \in X\}, \quad y \in R^m_+, \quad (2)$$

содержательно состоящую в минимизации суммар-

ных издержек $y^T F_0(x)$ производства из ограниченных объемов запасов ресурсов $P(y)$ при меняющейся рыночной стоимости $y \in R^m_+$ ресурсов и заданном технологическом процессе $F_1(x)$. Задача (2) может иметь общую форму, а именно

$$\min \{F(x, y) | F_1(x) \leq P(y), x \in X\}, \quad y \in R^m_+, \quad (3)$$

где целевая функция $F: R^n \times R^m_+ \rightarrow R$, характеризующая издержки производства, выпукла по переменной x для каждого фиксированного параметра y .

Наличие отображения $P: R^m_+ \rightarrow R^m_+$ в задаче (3) (или (2)) позволяет трансформировать одни единицы ресурсов в другие. Так, если y измеряется в денежных единицах, то $P(y)$ дает количественное представление ресурсов, если же $F_1(x)$ затраты производства, имеющие денежное выражение, то $P(y)$ имеет то же измерение, и тогда параметр y задает количество используемых ресурсов.

Для решения параметрической задачи (3) (или (2)) введем дополнительные условия в виде параметрической по параметру x' нелинейной задачи дополненности

$$P(y) \geq F_1(x'), \quad y \geq 0, \quad y^T (P(y) - F_1(x')) = 0, \quad (4)$$

где

$$x' \in \text{Arg min} \{F(x, y') | F_1(x) \leq P(y'), x \in X\},$$

или для задачи (2)

$$x' \in \text{Arg min} \{(y')^T F(x) | F_1(x) \leq P(y'), x \in X\}.$$

Введенное дополнительное условие $y^T (P(y) - F_1(x)) = 0$ есть условие дополняющей нежесткости для ограничений исходной задачи $F_1(x) \leq P(y)$ и внешних управляющих воздействий $y \geq 0$. Это условие придает внешним параметрам задачи (3) (или (2)) свойства ее двойственных переменных.

Вариационные неравенства со связанными ограничениями. Перепишем задачу дополненности (4) для (3) в несколько ином виде

$$\max \{y^T (F_1(x') - P(y)) | F_1(x') \leq P(y), y \in R^m_+\}, \quad (5)$$

где x' — решение соответствующей задачи

$$\min \{F(x, y') | F_1(x) \leq P(y'), x \in X\}. \quad (6)$$

В таком виде пару задач (5), (6) можно рассматривать как задачи со связанными ограничениями. Здесь в роли решаемой задачи выступает (6), а в роли связанного ограничения — оптимизационная задача (5).

Запишем эту задачу со связанными ограничениями в виде вариационного неравенства со связанными ограничениями. Введем для этого функции

$$H(v, w) = F(x_1, y_2) - \langle x_2, F_1(y_1) - P(x_2) \rangle,$$

$$G(v, w) = F_1(x_1) - P(y_2) + F_1(y_1) - P(x_2),$$

где $v = [x_1, y_1]$, $w = [x_2, y_2]$, $v, w \in \Omega = X \times R^m_+$. В терминах этих функций сформулируем новую задачу: найти вектор $v' \in \Omega$ такой, что

$$v' \in \text{Arg min} \{H(v', w) | G(v', w) \leq 0, w \in \Omega\}. \quad (7)$$

Покажем, что решение задачи (7) будет решением и исходной задачи (5), (6).

Задача (7) соответствует выполнению неравенства

$$F(x', y') + \langle y', F_1(x') - P(y') \rangle \leq F(x, y') + \langle y, (F_1(x') - P(y)) \rangle$$

для всех x и y , удовлетворяющих неравенству

$$F_1(x) - P(y') + F_1(x') - P(y) \leq 0, \quad x \in X, \quad y \in R_+^m.$$

Для дальнейших рассуждений важно отметить, что $G(v', v') \leq 0$; в исходных обозначениях это гарантирует выполнение следующего неравенства

$$F_1(x') - P(y') \leq 0. \quad (8)$$

Теперь зафиксируем y' , при этом неравенства останутся верными для всех $x \in X$, в этом случае задача преобразуется в следующую

$$F(x', y') \leq F(x, y')$$

для $x \in X$, удовлетворяющих неравенству

$$F_1(x) - F_1(x') \leq 0. \quad (9)$$

Очевидно, оптимальный вектор x' удовлетворяет этому неравенству, но оно отличается от неравенства в исходной постановке задачи и может задавать иную область поиска решений. В разрешении этой проблемы поможет неравенство (8), если построить двойное неравенство на основе (4) и (5), то получится $F_1(x) - F_1(x') \leq P(y')$. Это же неравенство гарантирует, что область поиска решений не больше, чем область поиска решений исходной задачи. Из чего можно сделать вывод, что, решая задачу (7), мы найдем решение задачи (6). Аналогичные рассуждения приводят к задаче (5), если закрепить x' .

В случае, когда целевая функция $H(v, w)$ дифференцируема, поставленную задачу можно представить в виде вариационного неравенства со связанными ограничениями

$$\langle \nabla_w H(v', v'), w - v' \rangle \geq 0, \quad \forall w \in \Omega, \quad G(v', w) \leq 0,$$

где $\nabla_w H(v, v) = \nabla_w H(v, w)|_{w=v}$.

Таким образом, получили модель несобственной задачи математического программирования (1) в виде вариационного неравенства со связанными ограничениями.

Двухшаговый экстраградиентный метод с памятью. Для решения вариационных неравенств в работе [3] предложен двухшаговый экстраградиентный метод. Для решения вариационных неравенств со связанными ограничениями применим его модификацию при условии дифференцируемости связанного ограничения. Схема метода будет выглядеть следующим образом

$$\bar{w}_k = (w_k + \alpha_k G(w_k, w_k))^+,$$

$$\bar{v}_k = P_\Omega(v_k - \alpha_k (\nabla_w H(v_k, v_k) + \nabla_w G(v_k, v_k) \bar{w}_k)),$$

$$\tilde{w}_k = (\bar{w}_k + \alpha_k G(\bar{w}_k, \bar{w}_k))^+,$$

$$\tilde{v}_k = P_\Omega(\bar{v}_k - \alpha_k (\nabla_w H(\bar{v}_k, \bar{v}_k) + \nabla_w G(\bar{v}_k, \bar{v}_k) \tilde{w}_k)),$$

$$w_{k+1} = (w_k + \alpha_k G(\tilde{w}_k, \tilde{w}_k))^+,$$

$$v_{k+1} = P_\Omega(v_k - \alpha_k (\nabla_w H(\tilde{v}_k, \tilde{v}_k) + \nabla_w G(\tilde{v}_k, \tilde{v}_k) \tilde{w}_k)),$$

где P_Ω и $(\cdot)^+$ — операторы проектирования на множества R_+^m и Ω соответственно.

Сформулируем условия сходимости этого метода в следующей теореме.

Теорема. Пусть для вариационного неравенства $\langle \nabla_w H(v', v'), w - v' \rangle \geq 0$ со связанным ограничением $G(v', w) \leq 0$ на выпуклом множестве Ω выполняются следующие условия:

- множество решений задачи не пусто;
- оператор $\nabla_w H(v', v')$ монотонный;
- функция $G(v, w)$ симметричная, дифференцируемая и выпуклая по w для любого $v \in \Omega$;
- выполняется равномерная оценка $\|y_k\| \leq C$ для всех $k \in N$;
- сужение $G(v, w)|_{w=v}$ на диагонали квадрата выпуклая функция.

Тогда существует последовательность $\{\alpha_k\}_{k \in N}$ параметров такая, что $\{v_k\}_{k \in N}$ сходится монотонно по норме к одному из равновесных решений.

Требования, необходимые для сходимости метода, на первый взгляд могут показаться обременительными, но заметим, что даже в такой общей формулировке условия в) и д) будут выполняться автоматически для исходной задачи. Условие г) выполняется в силу сходимости метода монотонно по норме как по прямым, так и по двойственным переменным. Условие б) гарантирует существование решения.

Обозначим через $[w_{k+1}, v_{k+1}] = \Phi([w_k, v_k])$ рекуррентные отношения построенного экстраградиентного метода для решения вариационных неравенств со связанными ограничениями. Тогда, при условии выполнения критерия одного направления,

$$\|\Phi([w_k, v_k]) - [w_k, v_k]\| \leq \sigma,$$

выполнение одной итерации экстраградиентного метода с памятью эффективнее исходного экстраградиентного метода и допускает реализацию алгоритма на многопроцессорных вычислителях [4].

Заключение. Математические модели сложных задач наряду с внутренними параметрами содержат внешние параметры, отвечающие за воздействия извне, влияющие на принимаемые решения. Приведенная в данной работе конструкция для решения несобственной задачи является новой и представляет большой интерес для использования, важным аспектом которого является применимость существующих методов для её разрешения и построение новых. Представление несобственной задачи в виде вариационного неравенства со связанными ограничениями позволяет применять к решению этой модели экстраградиентные методы с памятью, допускающие эффективную для параллельной реализации вычислительную схему.

Библиографический список

- Зыкина, А. В. Обратная дополняемость в модели управления ресурсами / А. В. Зыкина // Журнал вычислительной математики и математической физики. — 2008. — Т. 48. — № 11. — С. 1968 — 1978.
- Ерёмин, И. И. Противоречивые модели оптимального планирования / И. И. Ерёмин. — М.: Наука. — 1988. — 160 с.
- Зыкина, А. В. Двухшаговый экстраградиентный метод для вариационных неравенств / А. В. Зыкина, Н. В. Меленьчук // Известия вузов. Математика. — Казань: КГУ, 2010. — № 9. — С. 82 — 85.
- Запорожец, Д. Н. Распараллеливание экстраградиентных методов / Д. Н. Запорожец, В. С. Зыкин, А. В. Зыкина, Д. И. Куянов // Омский научный вестник. — 2011. — № 3(103). — С. 22 — 26.

ЗАПОРОЖЕЦ Дмитрий Николаевич, аспирант кафедры прикладной математики и фундаментальной информатики.

ЗЫКИНА Анна Владимировна, доктор физико-математических наук, профессор (Россия), заведующая

кафедрой прикладной математики и фундаментальной информатики.

Адрес для переписки: avzykina@mail.ru

Статья поступила в редакцию 28.08.2012 г.

© Д. Н. Запорожец, А. В. Зыкина

УДК 519.72

А. Н. ПОЛУЯНОВ

Омский государственный
технический университет

Омский филиал
Института математики
им. С. Л. Соболева Сибирского
отделения РАН

РАСЧЕТ ДИАГНОСТИЧЕСКОЙ ШКАЛЫ НА ГРАФИЧЕСКИХ ПРОЦЕССОРАХ

В работе рассматривается технология расчета медицинских диагностических шкал. Описана реализация параллельного алгоритма расчета шкалы с использованием графических процессоров (технология CUDA).

Ключевые слова: диагностические шкалы, параллельное программирование, CUDA.

Работа выполнена по проектам РФФИ 12-07-00066-а, 11-08-01349-а.

Введение. Анализ накопленной информации является актуальной проблемой для многих исследовательских и прикладных задач. Традиционным способом ее решения в настоящее время является технология оперативной аналитической обработки данных OLAP (online analytical processing). Основой OLAP-технологии является построение многомерных представления данных.

Можно выделить следующие шаги при работе с данными:

— исходные данные представлены в реляционном нормализованном виде, и к ним обеспечивается доступ по технологии OLTP (online transaction processing);

— пользовательское многомерное представление данных, реализующее технологию OLAP, обеспечивается инструментарием, преобразующим исходные данные в гиперкуб;

— пользовательское представление далее используется для анализа данных.

Реализация данной технологии с использованием последовательных алгоритмов подробно представлена в работах [1, 2].

Время работы алгоритмов формирования пользовательского представления данных и анализа данных можно значительно сократить, используя технологию CUDA [3, 4], предназначенную для разработки приложений, исполняемых на графических процессорах.

Графический процессор представляет собой вычислительное устройство, которое:

— является сопроцессором к центральному процессору (CPU);

— имеет собственную память;

— выполняет одновременно большое количество нитей (аналог потоков CPU).

Код приложения, разработанного по технологии CUDA, состоит как из последовательных, так и из

параллельных частей. Последовательная часть выполняется на CPU, а параллельная часть оформляется в виде функции ядра (kernel function) и выполняется на графическом процессоре.

В данной работе рассмотрен этап анализа данных, а именно реализация параллельного алгоритма расчета диагностической шкалы с использованием технологии CUDA.

Описание задачи. Традиционно для расчета диагностической шкалы [5] используется линейная комбинация N значимых параметров, называемая в литературе решающей функцией [6, 7]:

$$F(\mathbf{x}) = a_1 x_1 + a_2 x_2 + \dots + a_N x_N,$$

где $\mathbf{x} = (x_1, x_2, \dots, x_N)$ — вектор значений выделенных параметров (координат в пространстве параметров), $\mathbf{a} = (a_1, a_2, \dots, a_N)$ — веса выделенных параметров (коэффициенты).

Для значений функции $F(\mathbf{x})$ определяются границы (оценочная шкала):

$$g_0, g_1, \dots, g_K,$$

где K — количество групп объектов O_1, O_2, \dots, O_K . При условии, что $g_0 < g_1 < \dots < g_K$, определение принадлежности произвольного объекта o с вектором значений параметров \mathbf{x}' к группе O_j сводится к проверке выполнения неравенства:

$$g_{j-1} < F(\mathbf{x}') < g_j.$$

При выполнении равенства значения функции F какой-либо границе $F(\mathbf{x}') = g_j$ возникает ситуация неопределенности.

Для определения значений коэффициентов (a_1, a_2, \dots, a_N) и значений границ g_0, g_1, \dots, g_K в распознавании образов традиционно используются

обучающие выборки, заданные множеством групп объектов O_1, O_2, \dots, O_K . Пусть объект $o_{ij} \in O_i$ характеризуется вектором значений параметров: $x_{ij} = (x_{ij1}, x_{ij2}, \dots, x_{ijN})$. Функционалом риска выберем суммарное количество ошибок E при отнесении объекта к группе. Для текущих значений вектора (a_1, a_2, \dots, a_N) и границ g_0, g_1, \dots, g_K значение функционала риска вычисляется в следующем алгоритме:

```

E=0;
for  $i=1$  to  $K$ ;
for each  $o_{ij}$  in  $O_i$ ;
if  $F(x_{ij}) < g_{i-1}$  or  $F(x_{ij}) > g_i$  then  $E=E+1$ ;
endfor;
endfor.

```

В данном алгоритме ситуация неопределенности не считается ошибкой.

Таким образом, задача построения оценочной шкалы может быть записана в следующем виде:

$$E \rightarrow \min, g_0 < g_1 < \dots < g_K, -1 \leq a_i \leq 1, i=1, 2, \dots, N.$$

Ограничения на коэффициенты a_i реализуются за счет масштабирования. Общая схема решения задачи следующая:

1) выбирается начальное приближение для вектора весов $a^0 = (a_1^0, a_2^0, \dots, a_N^0)$;

2) циклически для $i=1, 2, \dots, N$ фиксируются все веса, кроме a_i , осуществляется спуск по координате a_i с определенным шагом, на каждом шаге выполняются пункты 3–6;

3) для текущего набора весов $a = (a_1, a_2, \dots, a_N)$ вычисляются значения $F_{ij} = F(x_{ij})$ для всех групп и всех объектов в группах с запоминанием номера группы для каждого значения;

4) значения F_{ij} сортируются по возрастанию;

5) $g_0 = \min(F_{ij}) - \epsilon$, $g_K = \max(F_{ij}) + \epsilon$, где ϵ — малая величина;

6) остальные значения границ g_1, \dots, g_{K-1} определяются перебором возможных вариантов при условии минимизации функционала риска и выполнения неравенств: $g_0 < g_1 < \dots < g_K$, причем значения выбираются посередине между соседними F_{ij} , чтобы не было ситуации неопределенности на обучающей выборке.

Последний шаг является наиболее трудоемким с вычислительной точки зрения.

Последовательная реализация представленных алгоритмов описана в работах [8, 9].

Параллельный алгоритм расчета шкалы. Специфика вычислительных устройств, построенных на базе графических процессоров, состоит в том, что программа для своего выполнения задействует как центральный процессор (CPU), так и графические процессоры (GPU). Поскольку на CPU выполняется последовательный код, то для ускорения работы программы основная часть вычислений должна выполняться на графических процессорах. Так же для ускорения работы программы необходимо оптимизировать работу с памятью.

При анализе последовательного алгоритма построения диагностической шкалы выявлено, что наибольшее время затрачивается на выполнение шага 6—96 % от общего времени работы программы, таким образом, по закону Амдала теоретически можно получить 25-кратный прирост производительности параллельной программы.

В рассматриваемом алгоритме на GPU параллельно выполняются наиболее трудоемкий шаг 6. Функция ядра (выполняемая на графическом процессоре) на входе получает массив с рассчитанными значениями решающей функции F_{ij} для каждого

объекта выборки при текущем наборе весов. Каждая нить, в зависимости от своего номера, рассчитывает значение функционала риска E для определенного набора границ g_0, \dots, g_K . Полученные результаты записываются в массив, в котором в качестве индекса выступает номер нити, а в качестве элемента — значения функционала риска, рассчитанного нитью для своего обрабатываемого набора границ. После завершения вычислений на GPU, CPU определяется минимальное значение функционала риска для текущего набора весов и выполняется следующий шаг алгоритма.

Поскольку при работе функции ядра происходят частые обращения к массиву значений решающей функции F_{ij} , данный массив целесообразно поместить в константную память графического процессора, обладающую меньшей латентностью по сравнению с глобальной памятью.

Параллельный алгоритм позволил значительно сократить временные затраты по обработке данных при построении диагностических шкал. Результаты расчетов показали в среднем десятикратное ускорение вычислений по сравнению с последовательным алгоритмом.

Отдельно хотелось бы остановиться на моменте оптимизации программного кода. Конечное время выполнения программы сильно зависит от архитектуры используемого графического процессора. Так, например, использование константной памяти на графических процессорах с архитектурой Tesla для хранения массива значений решающей функции (версия вычислителя до 1.3) позволило получить двукратное ускорение работы алгоритма. Анализ алгоритма в NVIDIA Visual Profiler показал, что при использовании константной памяти основное время работы графического процессора использовалось для выполнения инструкций функции ядра, в то время как без использования константной памяти основное время работы графического процессора уходило на запросы к глобальной памяти вычислителя. При запуске алгоритма на графических процессорах с архитектурой Fermi (версия вычислителя 2.0 и выше) значительных ускорений работы алгоритма при использовании константной памяти не наблюдалось, т.к. в архитектуре Fermi улучшена работа глобальной памяти GPU, добавлены дополнительные кэши.

При использовании в функции ядра операций с плавающей точкой скорость вычислений также определяется архитектурой GPU. В архитектуре Fermi, в отличие от Tesla, каждое ядро может производить вычисления с плавающей точкой, что существенно повышает скорость работы алгоритмов.

Также возможны некоторые сложности при распараллеливании алгоритма на несколько графических процессоров. Разработанный алгоритм использовался для вычислений на суперкомпьютере Tesla ОФ ИМ СО РАН, представляющим из себя гибридный кластер из трех узлов (два узла из трех графических процессоров с архитектурой Tesla, один узел из двух графических процессоров с архитектурой Fermi). Для распараллеливания алгоритма на несколько графических процессоров с архитектурой Tesla необходимо для каждого графического процессора создавать отдельный поток на CPU. На графических процессорах с архитектурой Fermi один поток CPU может использовать сразу несколько графических процессоров (функции ядра могут вызываться в цикле по доступным GPU, смена текущего GPU осуществляется путем вызова функции CudaSetDevice). Также на вычислителях с архитектурой Fermi все

графические процессоры одного узла имеют единое виртуальное адресное пространство, что позволяет осуществлять копирование данных глобальной памяти между различными GPU напрямую без использования оперативной памяти узла.

Представленные особенности технологии CUDA необходимо учитывать при разработке, отладке и оптимизации программного кода.

Заключение. Использование графических процессоров позволяет значительно повысить эффективность расчета диагностических шкал. Несмотря на некоторые сложности оптимизации параллельного алгоритма, связанные с активным развитием архитектуры графических процессоров и технологии их программирования, применение графических процессоров оправдало себя, позволив уменьшить на порядок временные затраты по обработке исходных данных.

В настоящее время продолжается работа по данной тематике. В перспективе планируется создание программного комплекса формирования и анализа многомерных данных с использованием графических процессоров.

Библиографический список

1. Зыкин, С. В. Формирование гиперкубического представления реляционной базы данных / С. В. Зыкин // Программирование. — 2006. — № 6. — С. 71–80.
2. Полуянов, А. Н. Автоматизация формирования гиперкубического представления данных / А. Н. Полуянов // Системы управления и информационные технологии. — 2008. — № 2 (32). — С. 289–293.
3. Боресков, А. В. Основы работы с технологией CUDA / А. В. Боресков, А. А. Харламов. — М. : ДМК Пресс, 2010. — 232 с.

4. Сандерс, Дж. Технология CUDA в примерах: введение в программирование графических процессоров / Дж. Сандерс, Э. Кэндрот. — М. : ДМК Пресс, 2011. — 232 с.

5. Александрович, Ю. С. Оценочные и прогностические шкалы в медицине критических состояний : сп. / Ю. С. Александрович, В. И. Гордеев. — Санкт-Петербург : Сотис, 2007. — 137 с.

6. Журавлёв, Ю. И. Об алгебраическом подходе к решению задач распознавания или классификации / Ю. И. Журавлёв // Проблемы кибернетики. — 1978. — Т. 33. — С. 5–68.

7. Лобов, Г. С. Метод адаптивного поиска логической решающей функции / Г. С. Лобов, В. М. Неделко, С. В. Неделко // Сибирский журнал индустриальной математики. — 2009. — Том XII, № 3(39). — С. 66–74.

8. Технология подготовки и анализа данных для построения медицинских оценочных шкал / С. В. Зыкин [и др.] // Информационные технологии. — 2010. — № 12. — С. 57–62.

9. Зыкин, С. В. Формирование представлений данных для построения медицинских диагностических шкал / С. В. Зыкин, П. Г. Редреев, А. К. Чернышев // Омский научный вестник. Сер. Приборы, машины и технологии. — 2011. — № 2(100). — С. 190–193.

ПОЛУЯНОВ Андрей Николаевич, кандидат технических наук, старший преподаватель кафедры «Прикладная математика и фундаментальная информатика» Омского государственного технического университета, научный сотрудник лаборатории «Методы представления и преобразования информации» Омского филиала Института математики им. С. Л. Соболева Сибирского отделения РАН.

Адрес для переписки: Andrey.Poluyanov@gmail.com

Статья поступила в редакцию 28.08.2012 г.

© А. Н. Полуянов

Книжная полка

Информационные системы и технологии в экономике и управлении : учеб. для вузов по специальности 080507 (061100) «Менеджмент организации» / В. В. Трофимов [и др.] ; под ред. В. В. Трофимова ; С.-Петерб. гос. ун-т экономики и финансов. — 3-е изд., перераб. и доп. — М. : Юрайт, 2011. — 1[521] с.—эл. опт. диск (DVD-ROM). — ISBN 978-5-9916-0919-7.

Настоящий учебник представляет собой обобщенный труд в области современных информационных систем и технологий, применяемых в экономике. Это универсальное издание для любых экономических специальностей. Материал учебника соответствует новым государственным стандартам и включает в себя не только обязательные разделы программы, но и дополнительный материал, поясняющий современное состояние дел в области создания и эксплуатации современных информационных систем и технологий, а также перспектив их развития.

Артемова, А. В. Решение оптимизационных задач при разработке средств вычислительной техники : учеб. пособие / А. В. Артемова, А. А. Колоколов, В. И. Потапов ; ОмГТУ. — Омск : Изд-во ОмГТУ, 2012. — 86 с. — ISBN 978-5-8149-1223-7.

В учебном пособии изложен ряд разделов по автоматизации проектирования сложных изделий, относящихся главным образом к разработке средств вычислительной техники, формированию производственных групп для создания программного обеспечения. Описывается подход к проектированию сложных изделий, основанный на использовании задач дискретной оптимизации с логическими ограничениями, приводятся соответствующие модели целочисленного линейного программирования, рассматриваются примеры задач оптимизации и способы их решения с помощью пакета MS Excel. В пособии содержатся также некоторые общие сведения о математическом моделировании и САПР.

РАЗРАБОТКА АРХИТЕКТУРЫ СИСТЕМЫ ОБНАРУЖЕНИЯ РАСПРЕДЕЛЕННЫХ СЕТЕВЫХ АТАК ТИПА «ОТКАЗ В ОБСЛУЖИВАНИИ»

В работе предложена архитектура специализированной системы, предназначенной для обнаружения сетевых DDoS-атак. В ходе первого этапа исследований был разработан модуль программно-аналитического комплекса, моделирующий сеть массового обслуживания, соответствующую исследуемой сети, с целью определения вероятности потерь заявок. В ходе второго этапа исследований на основе разработанной методики был реализован программно-аналитический комплекс, предназначенный для обнаружения распределенных сетевых атак типа «отказ в обслуживании».

Ключевые слова: информационная безопасность, распределенные сетевые атаки, обнаружение сетевых атак, отказ в обслуживании.

Введение. Одной из основных тенденций последних лет в сфере компьютерных преступлений является рост количества и сложности атак на доступность информации (ресурсов автоматизированной системы), как один из трех основных критериев, наряду с конфиденциальностью и целостностью информационной безопасности объекта. Данные атаки образуют класс атак типа «отказ в обслуживании» (DoS-атаки). Если атака выполняется одновременно с большого числа компьютеров, имеет место DDoS-атака (Distributed Denial of Service, распределенная атака типа «отказ в обслуживании»).

В целях минимизации последствий DDoS-атак, их обнаружение и классификация является крайне важной и вместе с тем сложной задачей. Основным способ распознавания DDoS-атаки заключается в обнаружении аномалий в структуре трафика. Традиционные механизмы обеспечения безопасности — межсетевые экраны и системы обнаружения вторжений — не являются эффективными средствами для обнаружения DDoS-атак и защиты от них, особенно атак трафиком большого объема [1]. Фундаментальной предпосылкой для обнаружения атак является построение контрольных характеристик трафика при работе сети в штатных условиях с последующим поиском аномалий в структуре трафика (отклонения от контрольных характеристик) [2]. Аномалия сетевого трафика — это событие или условие в сети, характеризующее статистическим отклонением от стандартной структуры трафика, полученной на основе ранее собранных профилей и контрольных характеристик. Любое отличие в структуре трафика, превышающее определенное пороговое значение, вызывает срабатывание сигнала тревоги.

Вместе с тем существующие методы обнаружения DDoS-атак, позволяющие эффективно распознавать DDoS-атаки транспортного уровня (SYN-флуд, UDP-флуд и другие), малоэффективны для обнаружения низкоактивных DDoS-атак прикладного уровня («медленный» HTTP GET флуд и «медленный» HTTP POST флуд) [3]. Данный класс DDoS-атак возник сравнительно недавно и на сегодняшний день пред-

ставляет основную угрозу доступности информации в распределенных компьютерных сетях [4]. Отличить трафик, генерируемый в ходе данных атак, от легального HTTP-трафика достаточно сложно, кроме того, каналы передачи данных практически не перегружаются. Данные атаки приводят к потерям запросов и ответов, т.е. фактическому отказу веб-серверов на основе Microsoft IIS, Apache и других систем. Кроме того, атака может быть адаптирована для воздействия на SMTP и даже на DNS-серверы.

Данные факты обуславливают необходимость разработки специализированной методики обнаружения низкоактивных распределенных атак прикладного уровня типа «отказ в обслуживании» в компьютерных сетях. Разрабатываемая методика обнаружения строится на основе оценки вероятности потерь пакетов или запросов прикладного уровня в компьютерной сети, при условии ее функционирования в стационарном режиме [5]. Достаточно часто для расчёта параметров потоков данных, а также для оценки вероятностей потерь пакетов в вычислительных сетях применяют математические модели в виде сетей массового обслуживания (СеМО). Использование СеМО в качестве модели вычислительной сети действительно дает возможность проводить анализ работы сети со сложной структурой и разнообразными сетевыми сервисами.

Архитектура модуля имитационного моделирования. В ходе первого этапа исследований на основе разработанной методики был реализован модуль программно-аналитического комплекса, моделирующий СеМО, соответствующую исследуемой сети, с целью определения вероятности потерь заявок (рис. 1). На вход системы поступает субстохастическая матрица, задающая топологию сети, интенсивности входных потоков и интенсивности обработки заявок в узлах.

Моделирующий модуль программного комплекса содержит объекты двух видов — источники заявок и узлы сети. Каждый из этих видов объектов (классов) имеет базовый класс, представляющий собой рабочий поток, каждый объект работает в отдельном

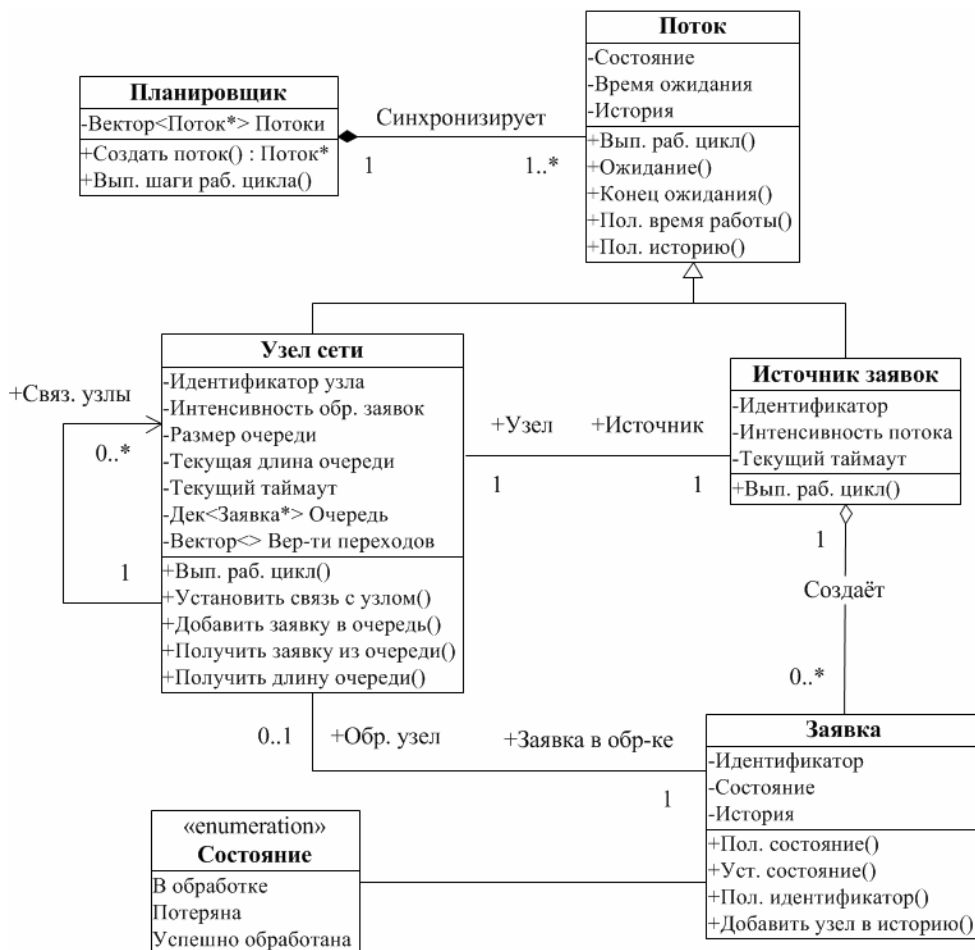


Рис. 1. UML-диаграмма классов моделирующего модуля

потоке в рамках одного процесса. Каждый источник заявок связан с одним из узлов СеМО. Источник заявок генерирует новые объекты заявок и добавляет их в очередь соответствующего узла. Начиная с момента запуска, узел проверяет наличие в очереди заявок. Если в очереди присутствует хотя бы одна заявка, она извлекается, после чего узел ожидает тайм-аут, имеющий экспоненциальное распределение, и далее случайным образом определяет заявку либо успешно обработанной, либо предпринимается попытка передать ее в очередь одного из смежных узлов. Заявка теряется, если при попытке добавления ее в очередь узла эта очередь полностью заполнена.

В процессе обработки заявки сохраняется вся информация о ее маршруте по сети массового обслуживания, что позволяет по завершении процесса моделирования точно рассчитать наблюдаемую частоту потери заявок. Функционирование объектов модели СеМО реализовано в параллельном режиме на основе библиотеки OpenMP, генерация псевдослучайных чисел осуществляется при помощи библиотеки Boost.

Архитектура комплекса обнаружения сетевых атак типа «отказ в обслуживании». В ходе второго этапа исследований для практического применения разработанной методики оценки вероятности потерь заявок был реализован программно-аналитический комплекс, предназначенный для обнаружения распространенных сетевых атак «отказ в обслуживании» [5]. Архитектура разработанного комплекса основана на модели «клиент – сервер», а сам комплекс конструктивно состоит из двух частей — клиентского и серверного модуля (рис. 2). Оба модуля реали-

зованы в качестве службы (сервиса) операционной системы Windows.

Клиентский модуль программно-аналитического комплекса при запуске:

- получает список доступных в системе сетевых интерфейсов;

- на каждом из интерфейсов запускается процесс захвата пакетов. Для этого фильтр пакетов конфигурируется таким образом, чтобы перехватывать фреймы Ethernet и протокола TCP с портами, соответствующими распространенным сетевым службам (NetBIOS, HTTP, FTP, SMTP, и т. д.);

- начинается отсчет времени. Когда время, прошедшее с начала захвата, превышает заданное значение t_{max} , захват трафика останавливается. Подсчитывается объем исходящего и входящего трафика, переданного другим узлам распределенной сети по контролируемым портам. Соответствующая информация передается на серверный модуль. Далее, начиная со второго шага, процесс повторяется.

Серверный модуль при запуске выполняет следующие действия.

1. Получает список доступных в системе сетевых адаптеров, и данные о них.

2. На каждом из адаптеров запускается процесс захвата пакетов. Для этого фильтр пакетов конфигурируется таким образом, чтобы перехватывать фреймы Ethernet, а вернее пакеты протокола TCP с портами, соответствующими распространенным сетевым службам.

3. Открывается порт для входящих соединений от клиентских модулей программно-аналитического комплекса.

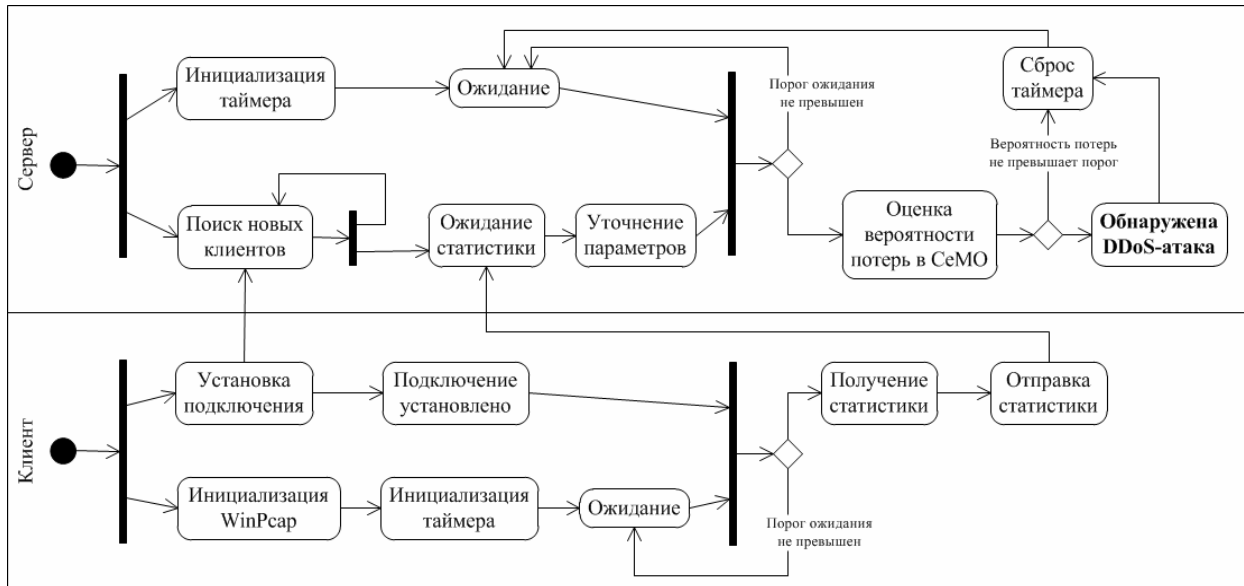


Рис. 2. UML-диаграмма деятельности комплекса обнаружения распределенных сетевых атак типа «отказ в обслуживании»

4. В случае входящего соединения от клиента:
- принимается информация о получателе трафика (пара адрес: порт) и количестве трафика, а также времени, за которое информация была собрана;
 - для каждой пары адрес: порт в сообщении от клиента уточняется полученное ранее среднее значение объема трафика, переданного текущим сокетом другим узлам по формуле:

$$Av_{k_1, k_2, t+t_{max}} = \frac{Av_{k_1, k_2, t} \cdot t + V_{k_1, k_2, t_{max}}}{t + t_{max}}.$$

Здесь:

- $Av_{k_1, k_2, t+t_{max}}$ — среднее значение объема трафика от сокета k_1 сокету k_2 к моменту времени $t+t_{max}$;
- t — время, прошедшее с начала наблюдений до предыдущего получения информации от клиентской части, t_{max} — время, за которое клиентская часть собирает данные о трафике сокета,
- $Av_{k_1, k_2, t}$ — средний объем трафика от сокета k_1 сокету k_2 за время, прошедшее с начала наблюдений до предыдущего получения информации от клиента,
- $V_{k_1, k_2, t_{max}}$ — объем трафика от сокета k_1 к сокету k_2 , прошедший за время t_{max} ;
- используя информацию о среднем значении объема трафика сокета k_1 сокету k_2 и суммарному входящему трафику в сокет k_1 , оцениваются параметры субстохастической матрицы, задающей топологию сети.

5. Когда истекло время t_{max} и захват трафика останавливается, подводятся итоги по объему трафика, отправленного текущим узлом другим узлам и характеристикам анализируемой сети массового обслуживания:

- используя субстохастическую матрицу, строится модель анализируемой CeMO и оцениваются параметры стационарного режима;
- используя параметры стационарного режима, строится цепь Маркова, соответствующая пути заявки (пакета) по CeMO;
- рассчитывается эволюция цепи Маркова и оценивается вероятность потерь;
- далее, начиная с четвертого шага, процесс повторяется.

Если в некоторый момент времени оценочная вероятность потерь превысила некоторое, наперед заданное администратором значение порога, система принимает вывод о реализации в распределенной компьютерной сети атаки типа «отказ в обслуживании».

Заключение. Таким образом, в результате работы построена программная реализация системы, предназначенной для обнаружения распределенных сетевых атак типа «отказ в обслуживании». Оценка эффективности разработанной системы обнаружения DDoS-атак в компьютерных сетях и её сравнительный анализ с другими подходами, методами и системами представляет сложную задачу. Основные трудности оценки и анализа заключаются в следующем:

- эффективность обнаружения атак типа «отказ в обслуживании» напрямую зависит от параметров работы сети в штатном режиме (загрузка сети, среднее значение потерь пакетов/запросов), при этом воспроизведение параметров работы для двух различных экспериментов не всегда является возможным;
- разрабатываемые методы и системы обладают рядом настраиваемых индивидуальных параметров (точность вычислений, значение порога принятия решения об обнаружении атаки), существенно влияющих на эффективность обнаружения атак и количество ложных срабатываний;
- кроме того, для каждой разновидности DDoS-атаки существует множество различных модификаций и параметров (интенсивность атаки, уникальные идентификаторы), которые также непосредственно влияют на эффективность обнаружения атаки.

Все вышеперечисленные факторы обуславливают отсутствие единого стандарта экспериментальных условий для оценки эффективности систем и методов обнаружения атак типа «отказ в обслуживании». Исследование и решение указанных задач будет произведено в ходе следующего этапа работы.

Библиографический список

1. Решение Cisco Systems «Clean Pipes» по защите от распределенных DOS-атак для операторов связи и их клиентов [Элек-

тронный ресурс]. — Режим доступа: http://www.cisco.com/web/RU/downloads/CleanPipes_rus.pdf, свободный (дата обращения: 01.08.2012).

2. Лобанов, В. Е. Архитектура системы защиты Грид от атак типа «отказ в обслуживании» и «распределенный отказ в обслуживании» / В. Е. Лобанов, Б. Н. Оныкий, А. А. Станкевичус // Безопасность информационных технологий. — 2010. — № 2010-3. — С. 136–139.

3. Обзор DDoS-атак во втором квартале 2011 года. — [Электронный ресурс]. — Режим доступа: http://www.securelist.com/ru/analysis/208050712/Obzor_DDoS_atak_vo_vtorom_kvartale_2011_goda (дата обращения: 01.08.2012).

4. Chee, W.O. Brennan, T. OWASP AppSec DC 2010. HTTP POST DDoS. — [Электронный ресурс]. — Режим доступа: https://www.owasp.org/images/4/43/Layer_7_DDOS.pdf (дата обращения: 01.08.2012).

5. Щерба, М. В. Система анализа устойчивости распределенных компьютерных сетей к атакам на «отказ в обслуживании» / М. В. Щерба // Омский научный вестник. Сер. Приборы, машины и технологии. — 2012. — № 1(107). — С. 295–298.

ЩЕРБА Евгений Викторович, кандидат технических наук, доцент кафедры «Комплексная защита информации».

Адрес для переписки: evscherba@gmail.com

ЩЕРБА Мария Витальевна, ассистент кафедры «Комплексная защита информации».

Адрес для переписки: mariz3@mail.ru

Статья поступила в редакцию 28.08.2012 г.

© Е. В. Щерба, М. В. Щерба

Книжная полка

Хорев, П. Б. Объектно-ориентированное программирование : учеб. пособие для вузов по направлению «Информатика и вычислительная техника» / П. Б. Хорев. — 3-е изд., испр. — М. : Академия, 2011. — 446 с. — ISBN 978-5-7695-8091-8.

Учебное пособие создано в соответствии с Федеральным государственным образовательным стандартом по направлению «Информатика и вычислительная техника» (квалификация «бакалавр»). Излагаются основные понятия технологии программирования. Большое внимание уделяется программированию для операционной системы Windows. Рассматриваются наиболее часто используемые в учебном процессе и разработке программного обеспечения системы программирования: Microsoft Visual C++, Borland C++ Builder и Borland Delphi.

Раннев, Г. Г. Интеллектуальные средства измерений : учеб. для вузов по направлению «Приборостроение» и специальностям «Информационно-измерительная техника и технологии», «Авиационные приборы и измерительно-вычислительные комплексы» / Г. Г. Раннев. — М. : Академия, 2011. — 262 с. — ISBN 978-5-7695-6469-7.

Рассмотрены проблемы интеллектуализации измерений, применение нейроструктуры в средствах измерений; измерительные базы знаний; особенности аппаратной и программной частей интеллектуальных средств измерений.

Морозов, В. К. Моделирование информационных и динамических систем : учеб. пособие для вузов по направлению подгот. «Автоматизация и управление» / В. К. Морозов, Г. Н. Рогачев. — М. : Академия, 2011. — 376 с. — ISBN 978-5-7695-4221-3.

Изложены основные понятия теории моделирования, приведена классификация моделей, рассмотрены модели непрерывных, дискретных и гибридных (агрегативных) систем. Рассмотрены особенности применения пакета MATLAB для решения круга задач моделирования систем. Содержатся примеры построения и применения моделей различных систем.

Головин, И. Г. Языки и методы программирования : учеб. для вузов по направлению 010400 «Прикладная математика и информатика» и 010300 «Фундаментальная информатика и информационные технологии» / И. Г. Головин, И. А. Волкова. — М. : Академия, 2012. — 303 с. — ISBN 978-5-7695-7973-8.

Учебник создан в соответствии с Федеральным государственным образовательным стандартом по направлению подготовки 010400 «Прикладная математика и информатика» (квалификация «бакалавр»). Рассмотрены основные парадигмы программирования: процедурная, объектно-ориентированная, функциональная. Особое внимание уделено семантике и прагматике языковых понятий, их связи с методами и технологией программирования. Материал представлен на примере современных языков индустриального программирования: C++, C#, Java. Отражены теоретические и практические вопросы реализации языков программирования.