

Метод и алгоритм построения допустимой последовательности изучения контента образовательного модуля

В.А. Фатхи, М.В. Гранков, М. Ал Зирки

Донской государственной технической университет

Аннотация: В статье рассмотрена проблема формального доказательства существования логически допустимой последовательности изучения контента образовательного модуля (дисциплины). При этом контент рассматривается как некоторая иерархическая модульная структура, состоящая из элементарных модулей. Каждый модуль такой структуры рассматривается как отображение входных понятий (термов) в выходные, которые осваиваются в данном модуле. Под входными терминами понимаются понятия, необходимые для освоения в данном модуле выходных термов. В статье вводится понятие дидактической зависимости между входными и выходными терминами. Данные зависимости являются формальным аппаратом описания ограничений в логической модели контента. Приведены правила преобразования дидактических зависимостей. Модульная структура контента и дидактические зависимости рассматриваются как формальный аппарат описания семантики предметной области. Показано, что введение множества неявных термов и правила преобразований дидактических зависимостей позволяют упростить логическую модель контента, не нарушая ее семантику.

Ключевые слова: модуль, учебно-методические комплексы, контент, дисциплина, алгоритм, терм, автоматизация.

Введение. Необходимость реформирования российской системы высшего профессионального образования уже несколько десятилетий осознана образовательным сообществом, работодателями и заинтересованными государственными структурами [1]. Одним из основных направлений реформирования является создание в РФ современных многоуровневых образовательных систем [2]. Проблемой проектирования таких систем является низкий уровень автоматизации формирования образовательного контента. В качестве примера такого контента в системе отечественного высшего образования (ВО) можно привести электронные учебно-методические комплексы (ЭУМК). Использование в образовании электронного контента расширяет возможности применяемых образовательных технологий и создает условия разработки систем,

адаптирующих образовательный контент к возможностям и целям обучающихся [3,4]. Автоматизированная система формирования электронного образовательного контента обязательно должна содержать подсистему разработки траектории (последовательности) освоения элементов этого контента. Для такой подсистемы важным является теоретическое обоснование существования логически связной (валидной) последовательности изучения образовательного контента ЭУМК. При наличии нескольких таких последовательностей (траекторий) наличие формальной процедуры проверки их валидности позволит разрабатывать автоматизированные системы выбора рациональной индивидуальной траектории, с учетом особенностей данного обучающегося [5,6].

Целью настоящей статьи является разработка теоретических основ метода и алгоритма теста на валидность траектории изучения контента модуля (дисциплины) образовательной программы.

Аналитическая модель. В статье использован подход формализации образовательных программ (ОП), изложенный в [7]. С точки зрения этого подхода целью изучения модуля ОП является освоение на заданном уровне логически связанных понятий и формирование определенных компетенций. Идентификаторы таких понятий будем называть терминами. Изучаемые в модуле термины назовем выходными. Для их освоения студент должен владеть другими терминами, которые будем называть входными. Для реализации данного подхода на уровне модуля представим каждый модуль ОП, так называемый d -модуль, как совокупность более простых дидактических единиц, каждая из которых так же обладает свойствами модульности [8,9]. Модули, образующие d -модуль, будем называть « π -модулями» (подмодулями). Освоение π -модуля так же связано с преобразованием его входных терминов в выходные.

Введем обозначения:

P – образовательная программа (ОП);

$$C = \{c_1, c_2, \dots, c_i, \dots, c_m\}; \quad (1)$$

– множество компетенций, осваиваемых по программе **P**;

$$T = \{t_1, t_2, \dots, t_k\}; \quad (2)$$

– множество термов, программы **P**;

$$\Pi = \{\pi_1, \pi_2, \dots, \pi_n\}; \quad (3)$$

– множество π -модулей рассматриваемого d -модуля.

Каждый π -модуль может иметь модульную структуру. Такой модуль будем называть сложным, а образующие его модули – дочерними. Модули, не имеющие дочерних, будем называть простыми. Для упрощения допустим, что все модули множества $\Pi = \{\pi_1, \pi_2, \dots, \pi_n\}$ – простые.

$$O_i \subset T; \quad (4)$$

– множество выходных термов, которые изучаются в модуле $\pi_i \in \Pi$.

$$I_i \subset T; \quad (5)$$

– множество входных термов, которые необходимы для освоения в модуле $\pi_i \in \Pi$ термов O_i .

Логично предположить, что любом π -модуле изучается не менее одного терма. В этом случае:

$$O_i \neq \emptyset, \forall i \mid i=1, 2, \dots, n. \quad (6)$$

$$I = (\cup I_i, \text{ где } i=1, 2, \dots, n), \quad (7)$$

– множество входных термов всех модулей множества Π .

$$O = (\cup O_i, \text{ где } i=1, 2, \dots, n), \quad (8)$$

– множество выходных термов всех модулей множества Π .

$$T_d = I \cup O, \quad (9)$$

– множество термов всех модулей множества Π .

При этом:

$$T_d = \{t_1, t_2, \dots, t_k\}, \text{ где } t_j \in T, j=1, 2, \dots, k. \quad (10)$$

Множество I_d входных термов d -модуля определим как множество всех термов, которые не изучаются в d -модуле, но используются в нем. В этом случае:

$$I_d = (I \setminus O). \quad (11)$$

Зависимости между термами. Для освоения обучающимся выходных термов некоторого π -модуля может потребоваться знание других термов. Другими словами, успешность освоения одних термов зависит от знания других. Как известно, аппарат «функциональных зависимостей» между атрибутами отношения является механизмом рационального проектирования реляционных баз данных [10]. Формализуем понятие зависимости между термами с целью получения аппарата формальной проверки логической связности π -модулей, образующих d -модуль.

Определение 1. Пусть дан модуль $\pi_i \in \Pi$, где Π – множество π -модулей, образующих d -модуль. I_i и O_i – множества входных и выходных термов модуля π_i соответственно. В этом случае будем говорить, что множество выходных термов (O_i) дидактически зависит от множества входных термов (I_i) и обозначать эту зависимость следующим образом:

$$\pi_i: I_i \Rightarrow O_i \quad (12)$$

Если существует дидактическая зависимость $\pi_i: I_i \Rightarrow O_i$, то будем говорить, что множество I_i дидактически определяет множество O_i . По аналогии с функциональными зависимостями [10], на множестве дидактических зависимостей d -модуля могут быть заданы правила их тождественного преобразования, соответствующие правилам Амстронга, определенных на множестве функциональных зависимостей отношения [10]. При описании правил тождественного преобразования дидактических зависимостей используем следующие обозначения:

A, B, C, D – подмножества множества термов T_d d -модуля;

\Rightarrow – дидактическая зависимость;

\rightarrow – импликация.

Правила преобразования дидактических зависимостей.

1. Рефлексивность:

$$(B \subset A) \rightarrow (A \Rightarrow B). \quad (14)$$

2. Дополнение:

$$(A \Rightarrow B) \rightarrow ((A \cup C) \Rightarrow (B \cup C)). \quad (15)$$

3. Транзитивность:

$$(A \Rightarrow B) \wedge (B \Rightarrow C) \rightarrow (A \Rightarrow C). \quad (16)$$

4. Самоопределение:

$$A \Rightarrow A. \quad (17)$$

5. Декомпозиция:

$$(A \Rightarrow (B \cup C)) \rightarrow (A \Rightarrow B) \wedge (A \Rightarrow C). \quad (18)$$

6. Объединение:

$$(A \Rightarrow B) \wedge (A \Rightarrow C) \rightarrow (A \Rightarrow (B \cup C)). \quad (19)$$

7. Композиция:

$$(A \Rightarrow B) \wedge (C \Rightarrow D) \rightarrow (A \cup C) \Rightarrow (B \cup D). \quad (20)$$

Включение во входное множество I_i всех термов, которые необходимы для описания в модуле π_i выходных термов O_i , может существенно усложнить модель d -модуля. Механизм уменьшения мощности множества входных термов каждого π -модуля, с сохранением всех имевшихся дидактических зависимостей, рассмотрим на примере. Пусть даны два модуля $\pi_1: I_1 \Rightarrow O_1$ и $\pi_2: I_2 \Rightarrow O_2$, где $I_1 = A$, $O_1 = B$, $I_2 = A \cup B$, $O_2 = C$. Докажем, что из множества I_2 можно вычесть множество A и при этом упрощенные дидактические зависимости будут эквивалентны исходным.

Дано: $(A \Rightarrow B) \wedge ((A \cup B) \Rightarrow C)$.

Доказать: $((A \Rightarrow B) \wedge (B \Rightarrow C)) \sim ((A \Rightarrow B) \wedge ((A \cup B) \Rightarrow C))$

Доказательство:

Для доказательства к выражению $(A \Rightarrow B) \wedge (B \Rightarrow C)$ последовательно применяем (15), (18) и (14) правила.

$$((A \Rightarrow B) \wedge (B \Rightarrow C)) \sim ((A \Rightarrow B) \wedge ((A \cup B) \Rightarrow (A \cup C))) \sim$$

$$((A \Rightarrow B) \wedge ((A \cup B) \Rightarrow A) \wedge ((A \cup B) \Rightarrow C)) \sim ((A \Rightarrow B) \wedge ((A \cup B) \Rightarrow C)).$$

Итак: $((A \Rightarrow B) \wedge (B \Rightarrow C)) \sim ((A \Rightarrow B) \wedge ((A \cup B) \Rightarrow C))$, что и требовалось доказать.

Приведенный пример иллюстрирует возможность использования правил (14-20) для сокращения мощности множества входных термов модуля с сохранением всех имеющихся дидактических зависимостей. Рассмотрим еще один механизм, позволяющий не указывать во множестве входных термов ряд используемых в модуле понятий.

Определение 2. Нечеткое множество термов, владением которыми, с заданным уровнем принадлежности, демонстрируют все обучающиеся по программе **P** на момент начала изучения этой программы, будем называть множеством неявных термов.

К множеству неявных термы могут быть отнесены, например, понятия, изучаемые в средней школе, освоение которых на заданном уровне гарантируются результатами соответствующих государственных итоговых аттестаций школьников. Отнесение понятий к множеству неявных термов позволяет исключить соответствующие термы из множества входных термов для модуля, в котором они используются. Приведем пример понятий из множества неявных термов d -модуля «Базы данных» образовательной программы по направления 010503. «Математическое обеспечение и администрирование информационных систем». При освоении π -модуля «Алгоритмы реорганизации хешированного файла» d -модуля «Базы данных» используются следующие понятия из средней школы: целые и натуральные числа, правильные и неправильные дроби, смешанное число, деление целых

чисел с остатком. Опираясь на результаты государственных итоговых аттестаций, проводимых в форме Единых Государственных Экзаменов, можно считать, что обучающиеся в вузе студенты владеют навыками использования дробей. В этом случае эти понятия можно отнести к множеству неявных термов. При этом отпадает необходимость отнесения соответствующих этим понятиям термов к множеству входных термов рассматриваемого π -модуля. Определение 2 вводит понятие множества неявных термов, но не формализует механизм отнесения термов к этому множеству. Для разработки такого формального, динамического механизма потребуются дополнительные исследования. В рамках настоящей работы будем считать, что отнесение понятий к множеству неявных термов выполняют ведущие преподаватели вуза, основываясь на своем опыте. В случае снижения уровня подготовки студентов, может потребоваться исключение из множества неявных термов ранее введенные в это множества понятия, и включение их во множества входных термов соответствующих модулей. С помощью входного контроля это позволит обнаружить студентов, не владеющих на достаточном уровне необходимыми понятиями и провести корректирующее обучение.

Важным свойством d -модуля является свойство связности.

Определение 3. Упорядоченное множество (кортеж):

$$\Psi_i = (\pi_{i1}, \pi_{i2}, \dots, \pi_{in}), \quad (21)$$

составленное из всех π -модулей множества $\Pi = \{\pi_1, \pi_2, \dots, \pi_n\}$ будем называть траекторией освоения образовательного контента d -модуля.

Исходя из Определения 3, для любой траектории освоения образовательного контента модуля выполняются следующие очевидные свойства:

$$(\forall \pi_{ij} \mid \pi_{ij} \in \Psi_i) \rightarrow ((\exists \pi_k \mid \pi_k \in \Pi) \wedge (\pi_{ij} = \pi_k)) \quad (22)$$

$$(\forall \pi_k \mid \pi_k \in \Pi) \rightarrow ((\exists \pi_{ij} \mid \pi_{ij} \in \Psi_i) \wedge (\pi_k = \pi_{ij})) \quad (23)$$

$$((\forall j | j=1,2,\dots,n) \wedge (\forall k | k=1,2,\dots,n) \wedge (j \neq k)) \rightarrow (\pi_{ij} \neq \pi_{ik}), \quad (24)$$

$$(\cup \{ \pi_{ij} \}, j=1,2,\dots,n) = \Pi, \quad (25)$$

Определение 4. Будем говорить, что траектории освоения образовательного контента $\Psi_i = (\pi_{i1}, \pi_{i2}, \dots, \pi_{in})$ обладает свойством валидности, если для модулей этой траектории выполняется следующее свойство:

$$(\pi_{ij} \in \Psi_i) \rightarrow (\Pi_{ij} \subset (\text{Id} \cup (\cup \text{Oik}, k=1,2,\dots,j-1))), | j=1,2,\dots,n). \quad (26)$$

Определение 5. Траекторию освоения образовательного контента d -модуля, обладающую свойством валидности, будем называть валидной или допустимой траекторией.

Свойства (21) – (26) допустимой траектории Ψ_i можно интерпретировать следующим образом: если траектория Ψ_i является валидной (допустимой), то для любого ее модуля π_{ij} все входные термы изучались в предшествующих модулях, либо принадлежат множеству Id входных термов d -модуля.

Определение 6. Будем говорить, что d -модуль обладает свойством связности, если для него существует хотя бы одна валидная траектория Ψ_i .

Определение 7. Подмножество, состоящее из всех таких термов множества Td d -модуля, которые дидактически зависят от входных термов Id этого модуля, будем называть замыканием входных термов d -модуля и обозначать Id^+ .

Правило (17) обеспечивает выполнение следующего условия:

$$\text{Id} \subset \text{Id}^+. \quad (27)$$

Из Определения 7 следует, что Id^+ - это максимальное подмножество множества всех термов Td d -модуля, которые дидактически зависят от множества Id . В этом случае выполняется следующее условие:

$$(\text{Id} \subset \text{Td}) \wedge (\text{Id}^+ \subseteq \text{Td}) \wedge (\text{Id} \Rightarrow \text{Id}^+) \wedge (\forall \pi_k | (\pi_k \in \Pi) \wedge (\{ \pi_k \} \not\subset \text{Id}^+) \rightarrow \neg(\text{Id} \Rightarrow \{ \pi_k \})) \quad (28)$$

Алгоритм построения замыкания входных термов.



Дано:

$\Pi = \{\pi_1, \pi_2, \dots, \pi_n\}$ - множество простых π -модулей, образующих d -модуль, где $|\Pi| = n$ – мощность множества Π .

Id – множество входных термов d -модуля.

I_i и O_i – множества входных и выходных термов модуля $\pi_i \in \Pi$ соответственно.

Построить:

Id^+ – замыкание множества Id .

В тексте алгоритма символами /* */ отмечены комментарии.

Алгоритм:

1. $Id^+ := Id$; /* За начальное значение множества Id^+ принять множество Id */
 2. СОЗДАТЬ ТРАЕКТОРИЮ НУЛЕВОЙ ДЛИНЫ Ψ ;
 3. ЦИКЛ_ПОКА ИСТИНА; /* Итерационный цикл */
 4. $i := 0$; /* Начальное значение переменной цикла по элементам Π */
 5. ФЛАГ_ИЗМЕНЕНИЯ_ТРАЕКТОРИИ := ЛОЖЬ;
 6. ЦИКЛ_ПОКА $i < n$; /* Цикл по номеру i модуля π_i во множестве Π */
 7. $i := i + 1$;
 8. ЕСЛИ $(I_i \subseteq Id^+) \wedge (O_i \not\subseteq Id^+)$ ТО /* Входные термы модуля π_i принадлежат замыканию Id^+ и выходные термы π_i еще не содержатся в Id^+ */
 9. $Id^+ := Id^+ \cup O_i$; /* Добавить множество O_i выходных термов модуля π_i в замыкание Id^+ */
 10. В ТРАЕКТОРИЮ Ψ МОДУЛЬ ДОБАВИТЬ π_i ;
 11. ФЛАГ_ИЗМЕНЕНИЯ_ТРАЕКТОРИИ := ИСТИНА;
 12. ВСЕ_ЕСЛИ;
 13. ВСЕ_ЦИКЛ;
 14. ЕСЛИ ФЛАГ_ИЗМЕНЕНИЯ_ТРАЕКТОРИИ ТО
 15. ИНАЧЕ
-

16. ПРЕРВАТЬ_ЦИКЛ;
17. ВСЕ_ЕСЛИ;
18. ВСЕ_ЦИКЛ;

Предлагаемый алгоритм строит множество термов Id^+ , которое может освоить обучающийся при изучении d -модуля, если он владеет терминами Id . После завершения алгоритма множество Id^+ содержит все термы (включая множество Id), которыми будет владеть обучающийся, при успешном освоении d -модуля.

В пункте 8 проверяется условие, что во множестве π -модулей рассматриваемого d -модуля обнаружен новый модуль π_i , который может быть освоен обучающимся. В пункте 9 выходные термы этого модуля добавляются во множество Id^+ , а в пункте 10 в возможную траекторию освоения d -модуля добавляется новый модуль π_i . В пункте 11, переменная ФЛАГ_ИЗМЕНЕНИЯ_ТРАЕКТОРИИ устанавливается в состояние «ИСТИНА». Итерационная обработка алгоритмом множества π -модулей Π продолжается до тех пор, пока после очередной итерации значение переменной ФЛАГ_ИЗМЕНЕНИЯ_ТРАЕКТОРИИ останется в состоянии «ЛОЖЬ».

Алгоритм проверки d -модуля на связность.

Дано:

$\Pi = \{\pi_1, \pi_2, \dots, \pi_n\}$ - простые π -модули, образующие d -модуль.

T_d – множество всех термов d -модуля.

Id – множество входных термов d -модуля.

I_i и O_i – множества входных и выходных термов модуля $\pi_i \in \Pi$ соответственно.

Проверить:

Свойство связности d -модуля.

Алгоритм проверки d -модуля на связность использует алгоритм построения замыкания множества входных термов d -модуля;

1. ПОСТРОИТЬ_МНОЖЕСТВО Id^+ ;
2. ЕСЛИ $Td=Id^+$ ТО /* Проверяется условие, что все термы d -модуля могут быть освоены обучающимся*/
3. СВЯЗНОСТЬ_МОДУЛЯ:=ИСТИНА;
4. Ψ -ВАЛИДНАЯ ТРАЕКТОРИЯ;
5. ИНАЧЕ
6. СВЯЗНОСТЬ_МОДУЛЯ:=ЛОЖЬ;
7. ВСЕ_ЕСЛИ;

В пункте 1 алгоритма выполняется построение замыкания множества входных термов Id^+ d -модуля. При этом выполняется попытка построения Ψ - валидной траектории изучения всех π -модулей рассматриваемого d -модуля. В пункте 2 алгоритма проверяется условие, что множество всех термов d -модуля совпадает с замыканием его входных термов. Равенство этих множеств ($Td=Id^+$) обеспечивает возможность освоение обучающимся всех термов d -модуля и существование валидной траектории. Следовательно, d -модуль обладает свойством связности.

Выводы. Использование иерархического модульного представления структуры образовательного контента позволило разработать инвариантные механизмы описания логической модели любого сложного модуля (дисциплины) Электронного Учебно-Методического Комплекса. Введение понятия дидактической зависимости между термами модуля, правила их преобразования, замыкание множества входных термов являются теоретической основой метода тестирования образовательного контента на логическую связность. Предложенные в статье метод и алгоритмы позволяют формально строить валидные траектории освоения студентами учебных

материалов и применяться в автоматизированных и адаптивных обучающих системах.

Литература

1. Бородина, Н.А., Богданова, И.Б. Особенности осуществления государственной политики в области информатизации образования в современной России // Инженерный вестник Дона, 2012, №1 URL: ivdon.ru/magazine/archive/n1y2012/635.

2. Моисеенко, Н.А. Трансформационное обучение и холистический подход в информационно-образовательной среде технического вуза // Инженерный вестник Дона, 2013, №4 URL: ivdon.ru/ru/magazine/archive/n4y2013/2034.

3. Разыграева, В.А. Автоматизация процесса адаптивного электронного обучения с учетом функционального состояния обучающегося: автореф. дис. ... канд. техн. наук: 05.13.06 / Разыграева Вероника Александровна. – Санкт-Петербург, 2011. – 19 с.

4. Иванов И.В., Косоногова М.А. Кибернетическая модель адаптивного электронного учебника с использованием методологии Г.Раша.//Приборы и системы. Управление, контроль, диагностика. — 2014. — №9 —с. 73-78.

5. Белоус, В.В. Нечеткие модели и методы управления образовательной траекторией в системе переподготовки персонала промышленных предприятий: автореф. дис. ... канд. техн. наук: 05.13.06 / Белоус Валентина Владимировна. – Москва, 2009. – 22 с.

6. Норенков, И.П., Соколов Н.К. Синтез индивидуальных маршрутов обучения в онтологических обучающих системах / И.П. Норенков, Н.К. Соколов // Информационных технологии. – 2009. – №3. – с. 74–77.

7. Ал Зирки М., Гранков М.В. Формализация критериев успешного освоения студентами образовательной программы // Технические науки.



Вестник Донского государственного технического университета.-2012 г. №3 с. 51-55.

8. Строганов, В. Ю. Комплексная автоматизация и моделирование адаптивных процессов тестового контроля и обучения с системе аттестации и подготовки кадров предприятий промышленности и транспортного комплекса: дис. ...докт. техн. наук: : 05.13.06 / Строганов Виктор Юрьевич. – Москва, 2004. – 396 с.

9. Lin, H.-Y. Design and Implementation of an Object Oriented Learning Activity System / H.-Y. Lin, S.-S. Tseng, J.-F. Weng, J.-M. Su // Educational Technology & Society. – 2009. – № 12(3). – pp. 248–265.

10. Date, C.J. An introduction to database systems. Eighth edition. Addison-Wesley Longman, Inc, 2004. p. 1034

References

1. Borodina, N.A., Bogdanova, I.B. Inzhenernyj vestnik Dona (Rus), 2012, №1 URL: ivdon.ru/magazine/archive/n1y2012/635.

2. Moiseenko, N.A. Inzhenernyj vestnik Dona (Rus), 2013, №4 URL: ivdon.ru/ru/magazine/archive/n4y2013/2034.

3. Razygraeva, V.A. Avtomatizacija processa adaptivnogo jelektronnogo obuchenija s uchetom funkcional'nogo sostojanija obuchajushhegosja. [Automating process of adaptive e-learning, based on the functional state of learner] Avtoref. dis. ... kand. tehn. nauk: 05.13.06. Razygraeva Veronika Aleksandrovna. Sankt-Peterburg, 2011 .19 p.

4. Ivanov I.V., Kosonogova M.A. Kiberneticheskaja model' adaptivnogo jelektronnogo uchebnika s ispol'zovaniem metodologii G.Rasha. Pribory i sistemy. Upravlenie, kontrol', diagnostika. 2014. №9 pp. 73-78.

5. Belous, V.V. Nechetkie modeli i metody upravlenija obrazovatel'noj traektoriej v sisteme perepodgotovki personala promyshlennyh predpriyatij. [Fuzzy

models and methods management of educational trajectory in system retraining personnel of industrial enterprises] avtoref. dis. ... kand. tehn. nauk: 05.13.06.

Belous Valentina Vladimirovna. Moskva, 2009. – 22 p.

6. Norenkov, I.P., Sokolov N.K. Informacionnyh tehnologii. 2009. №3. pp. 74–77.

7. Al Zirki M., Grankov M.V. Tekhnicheskie nauki. Vestnik Donskogo gosudarstvennogo tekhnicheskogo universiteta. 2012 g. №3. pp.51-55.

8. Stroganov, V. Ju. Kompleksnaja avtomatizacija i modelirovanie adaptivnyh processov testovogo kontrolja i obuchenija s sisteme attestacii i podgotovki kadrov predpriyatij promyshlennosti i transportnogo kompleksa. [Comprehensive of automation and modeling of adaptive process control test and training system certification and training personal of industrial and transport complex personnel]. dis. ...dokt. tehn. Nauk. 05.13.06. Stroganov Viktor Jur'evich. Moskva, 2004. 396 p.

9. Lin, H.-Y. Design and Implementation of an Object Oriented Learning Activity System. H.-Y. Lin, S.-S. Tseng, J.-F. Weng, J.-M. Su. Educational Technology & Society. 2009. № 12(3). pp. 248–265.

10. Date, C.J. An introduction to database systems. Eighth edition. Addison-Wesley Longman, Inc, 2004. p. 1034