УДК 007:519.816

ЛОГИКА ВЕТВЯЩЕГОСЯ ВРЕМЕНИ И ВОЗМОЖНОСТИ ЕЕ РЕАЛИЗАЦИИ *

А.П. Еремеев[1], И.Е. Куриленко[2]

Рассматриваются основные положения и возможности реализации темпоральной логики ветвящегося времени в плане ее использования в интеллектуальных системах поддержки принятия решений реального времени.

Введение

В работе [Еремеев, 2006] рассмотрены различные виды логики ветвящегося времени в плане их использования для поиска решений в интеллектуальных системах поддержки принятия решений реального времени (ИСППР РВ). Отмечалось, что ветвящаяся структура времени (ветвящееся время) в отличие от линейной допускает множественность будущего и/или прошлого. Такое время соответствует концепции «возможных миров» Крипке (Kripke) и позволяет моделировать различные пути развития ситуаций (различные параллельные миры) [Карпов, 2010].

Модели (логики) ветвящегося времени целесообразно использовать в ИСППР РВ при прогнозировании последствий принимаемых решений или развития некоторой ситуации в условиях дефицита времени.

1. Пропозициональная темпоральная логика ветвящегося времени

Рассмотрим наиболее простую относительно сложности реализации (что весьма существенно для ИСППР РВ) и в то же время достаточно выразительную ***пропозициональную темпоральную логику ветвящегося времени*** *(BPTL - Branching-Time Propositional Temporal logic)*, описанную в [Torsun, 1998] и являющуюся расширением ***пропозициональной темпоральной логики*** *(PTL)*. PTL является модальной темпоральной

* Работа выполнена при финансовой поддержке РФФИ (проект № 08-01-90437)

[1] *111250, Москва, Красноказарменная 14, МЭИ (ТУ), eremeev@appmat.ru*

[2] *111250, Москва, Красноказарменная 14, МЭИ (ТУ), ivan@appmat.ru*

логикой, построенной на основе классической логики с добавленными модальными операторами для дискретного линейного времени.

Синтаксис PTL задается следующим образом.

Языком L_P PTL является счетное множество пропозициональных символов *p, q, r, s,… Формулы* строятся, используя следующие символы:

- множество пропозициональных символов языка L_P;
- классические связки: $T, F, \neg, \wedge, \vee, \Rightarrow, \Leftrightarrow$;
- темпоральные операторы будущего времени: унарные - O, ◊, □; бинарные – *U,W*;
- темпоральные операторы прошлого времени: унарные - ⊗, ●, ♦, ■; бинарные – *J, Z*.

Множеством правильно построенных формул (ппф) PTL являются:

- все пропозициональные символы языка L_P есть ппф;
- если *A* и *B* – ппф, то ппф также будут и следующие выражения: *T, F;* $\neg A$, $A\wedge B$, $A\vee B$, $A\Rightarrow B$, $A\Leftrightarrow B$; *OA, ◊A, □A, AUB, AWB; ●A, ⊗A, ♦A, ■A, AJB, AZB*.

Интуитивный (неформальный) смысл модальных операторов следующий. Унарных: O -следующий (next), ● – прошлый (last), ⊗ - прошлый (last), □ - всегда в будущем (always in the future), ■ - всегда в прошлом (always in the past), ◊ - иногда в будущем (sometime in the future), ♦ - иногда в прошлом (sometime in the past); бинарных: *U* - до тех пор, пока (until), *W* - пока не (unless), *J* - с тех пор, как (since), *Z* – с тех пор, как (zince).

Если *A* и *B* являются пропозициональными формулами, то интуитивный смысл модальных формул определяется следующим образом: «Ппф O*A* истинна в данный момент (в данном состоянии), если ппф *A* истинна в следующий момент (состояние); ппф □*A* («always - всегда» *A*) истинна в данный момент, если и только если *A* истинна во все будущие моменты (состояниях, включая текущее); ппф ◊*A* («eventually - в конце концов» *A*) истинна в данный момент, если и только если *A* истинна в некоторый будущий момент. Строгая until ппф *AUB* истинна в данный момент (состояние), если и только если ппф *B* в конце концов будет истинна, т.е. в момент $s>n$, где n – текущий момент, и ппф *A* истинна для всех моментов *t*, таких что $n\leq t<s$.

Оператор *W* есть слабая версия оператора *U*, когда не гарантируется истинность ппф *B* в некоторый будущий момент. Темпоральные операторы прошлого времени определяются как строгая версия прошлого времени для соответствующих операторов будущего («будущих» двойников), т.е. прошлое не включает настоящий момент (состояние).

Семантика PTL определяется следующим образом.

Для задания семантики PTL используется семантика возможных миров Крипке. Возможный мир рассматривается как множество состояний во времени, связанных темпоральными отношениями из множества допустимых отношений *R*. Формально мир задается парой $A=(S,R)$, где S – непустое множество возможных состояний, $R \subseteq S \times S$ – бинарное отношение.

Рассматривая L_P как множество атомных утверждений, модель мира можно определить как $M=(R,S,V)$, где V – функция означивания (valuation function), задающая отображение $V: S \times L_P \rightarrow \{T,F\}$, т.е. вычисляющая пропозициональное значение для каждого состояния $s \in S$. Вводя различные ограничения на отношение *R*, получают различные модельные структуры. Например, если ввести ограничение антирефлексивности (<), то получим дискретную модель. Для дискретных линейных моделей множество *S* можно рассматривать как последовательность состояний, *R* - как отношение следования (next) или преемника (successor). Интерпретация задается парой $\langle M,i \rangle$. где *M* - модель, а *i* – целое число, индексирующее состояния $s_i \in S$ в модели.

Семантика для темпоральной ппф задается с использованием отношения $\models$ между интерпретацией и ппф. Так утверждение $\langle M,i \rangle \models A$ означает, что ппф *A* интерпретируется в модели *M* как состояние с индексом i. Семантика для неограниченной дискретной линейной темпоральной логики задается следующим образом:

- Для утверждений:

$\langle M,i \rangle \models A \leftrightarrow V(i,p)=T$ для $p \in L_P$, где символ $\leftrightarrow$ используется как сокращение для «если и только если».

- Для связок:

$\langle M,i \rangle \models T$

$\langle M,i \rangle \models \neg A \leftrightarrow not\ \langle M,i \rangle \models A$

$\langle M,i \rangle \models A \wedge B \leftrightarrow \langle M,i \rangle \models A$ и $\langle M,i \rangle \models B$

$\langle M,i \rangle \models A \vee B \leftrightarrow \langle M,i \rangle \models A$ или $\langle M,i \rangle \models B$.

Семантика для связок $\Rightarrow$ и $\Leftrightarrow$ определяется посредством $\neg$ и $\vee$.

- Для темпоральных модальных операторов:

$\langle M,i \rangle \models \bigcirc A \leftrightarrow \langle M,i+1 \rangle \models A$

$\langle M,i \rangle \models \Box A \leftrightarrow (\forall j \geq i)(\langle M,j \rangle \models A)$

$\langle M,i \rangle \models \Diamond A \leftrightarrow (\exists j \geq i)(\langle M,j \rangle \models A)$

$\langle M,i \rangle \models AUB \leftrightarrow (\exists j \geq i)(\langle M,j \rangle \models B$ и $(\forall k)((i \leq k < j) \rightarrow \langle M,k \rangle \models A)$

$\langle M,i \rangle \models \bullet A \leftrightarrow i=0$ или $\langle M,i-1 \rangle \models A$

$\langle M,i \rangle \models \otimes A \leftrightarrow i>0$ и $\langle M,i-1 \rangle \models A$

$\langle M,i \rangle \models \blacksquare A \leftrightarrow (\forall j, 0 \leq j < i)(\langle M,j \rangle \models A)$

$\langle M,i \rangle \models \blacklozenge A \leftrightarrow (\exists j, 0 \leq j < i)(\langle M,j \rangle \models A)$

$\langle M,i\rangle \models A\ J\ B \leftrightarrow (\exists j, 0 \leq j < i)(\langle M,j\rangle \models B$ и $(\forall k)((j \leq k < i) \rightarrow \langle M,k\rangle \models A)$

$\langle M,i\rangle \models A\ Z\ B \leftrightarrow \langle M,i\rangle \models A\ J\ B$ или $\langle M,i\rangle \models \blacksquare A$.

Интуитивная семантика для модальных операторов определяется следующим образом:

- O*A* - ппф O*A* истинна в текущем состоянии, если и только если ппф *A* истинна в следующем состоянии;
- $\Box A$ – ппф $\Box A$ истинна в текущем состоянии, если и только если ппф *A* истинна во всех состояниях в будущем;
- $\Diamond A$ - ппф $\Diamond A$ истинна в текущем состоянии, если и только если ппф *A* истинна в каком-либо состоянии в будущем;
- A*U*B – строгая until ппф A*U*B истинна в текущем состоянии s_i, если и только если ппф *B* будет в конце концов истинна, например, в состоянии s_j, j>i, и ппф *A* истинна для всех состояний s_k, $i \leq k < j$;
- *AWB* – слабая until ппф *AWB* (называемая unless ппф) означает, что *A* должна быть истинной до состояния, когда *B* станет истинной.

Различие между ппф *AUB* и *AWB* состоит в том, что для истинности *AUB* требуется когда-нибудь в будущем истинность *B*, а для *AWB* этого не требуется. Модальные операторы прошлого времени (past-time modal operators) определяются как зеркальное отображение модальных операторов будущего. Однако, операторы ■ («всегда в прошлом») и ♦ («когда-нибудь в прошлом») интерпретируются как строгие, что означает, что текущий индекс не включается в их определение. Кроме того предполагается, что время в прошлом ограничено и имеет начало. Также различаются два других оператора прошлого времени ⊗ и ●. Ппф ⊗*A* является ложной, в то время как ппф ●*A* истинна, только когда интерпретация дается в начале времени. Ппф ●*false* истинна, когда интерпретация дается для начала времени, в остальное время она ложна. Следовательно ¬●*true* можно использовать для определения начала (начального момента) времени. Справедливо: $\otimes\neg A \equiv \neg\bullet A$.

Понятия *выполнимости (satisfaction)* и *общезначимости (validity)* ппф определяются следующим образом:

- ппф *A* *выполнима в модели M* в состоянии s_i, если и только если $\langle M,i\rangle \models A$; и ппф *A* выполнима в модели *M*, если и только если *A* выполнима в *M* в состоянии s_0;
- ппф *A* *общезначима* ($\models A$), если и только если *A* истинна во всех моделях *M*.

Доказано, что PTL разрешима ($M \models A$) [Torsun, 1998].

Аксиоматика PTL. Аксиоматическая система для PTL включает следующие элементы:

- (A_0) – все элементы общезначимой схемы пропозициональной логики;
- аксиомы для оператора O:

(A_1): $O(A\Rightarrow B)\Rightarrow(OA\Rightarrow OB)$ - схема для нормальной логики *K*;

(A_2): $\neg OA \equiv O\neg A$ - существование и уникальность преемника;

- аксиомы для операторов $\Box$ и $\Diamond$:

(A_3): $\Diamond \equiv \neg\Box\neg A$ - аксиома двойственности;

(A_4): $\Box (A\Rightarrow B)\Rightarrow(\Box A\Rightarrow\Box B)$ - *K* аксиома;

- аксиомы, описывающие связь между операторами $\Box$ и O:

(A_5): $\Box A\Rightarrow(A\wedge OA\wedge O\Box A)$;

(A_6): $\Box(A\Rightarrow OA)\Rightarrow(A\Rightarrow\Box A)$;

- аксиомы, характеризующие оператор *U*:

(A_7): $AUB \Rightarrow(B\vee(A\wedge O(AUB)))$;

(A_8): $(F\wedge\Box(F\Rightarrow(B\vee(A\wedge OF))))\Rightarrow AUB$, где *F* – произвольная ппф.

Правилами вывода являются правила modus ponens (MP) и необходимости (N): (MP): если $\models A$ и $\models A\Rightarrow B$, то $\models B$; (N): если $\models A$, то $\models\Box A$. Система аксиом для линейной PTL *непротиворечива (sound)* и *полна (complete)* [Torsun, 1998].

В линейной дискретной логике PTL моделью времени является упорядоченная последовательность натуральных чисел, т.е. каждое состояние имеет одного и только одного преемника. В BPTL для каждого состояния не обязательно наличие единственного преемника и может быть множество возможных путей (последовательностей состояний) из любого данного состояния и, следовательно, возможно несколько различных «будущих». Моделью времени в BPTL является инфинитное дерево, но каждая вершина которого имеет финитное, ненулевое число преемников. Вершина дерева может рассматриваться как возможное состояние, а ветвь или путь – как история возможного мира. Допускается существование лишь одного прошлого, но разрешается будущему быть открытым.

Чтобы специфицировать состояния дерева, для которых ппф является истинной, необходимо идентифицировать путь, содержащий это состояние, и позицию состояния на этом пути. Таким образом делается различие между формулой пути (path formulae) и формулой состояния (state formulae). Формула пути является формулой PTL, а формула состояния указывает тот путь, на котором следует интерпретировать формулу пути, т.е. эта формула интерпретируется на данном состоянии пути. Например, если *p* есть утверждение и *s* есть состояние, то формула $\exists\Box p$ будет истинной в интерпретации «если существует путь, порождаемый в *s* и на котором все состояния удовлетворяют p». Обобщенно, если *p* есть формула пути, то $\forall p$ и $\exists p$ – формулы состояния. В состоянии s формула $\forall p$ интерпретируется как истинная, если формула p истинна для всех путей, ветвящихся (исходящих) из s, а формула $\exists p$ интерпретируется как истинная, если p истинна хотя бы на одном пути, исходящим из s.

Используя PTL, определены синтаксис и семантика BPTL. Формальная семантика BPTL определяется в терминах модельной структуры $M = (S,R,V)$, где S, R и V определяются аналогично PTL. Концепция ветвящегося времени требует введения условия линейности влево (в прошлое) и транзитивности R. Ппф BPTL являются формулами состояния, а формулы пути - вспомогательные объекты, вводимые для того, чтобы облегчить задание (выражение) семантики формул состояния.

Ппф является *непротиворечивой (consistent)*, если она истинна в некотором состоянии при некоторой интерпретации, и *общезначимой (valid)*, если она истинна в каждом состоянии при каждой интерпретации. Отрицанием общезначимой ппф является *невыполнимая (inconsistent)* ппф. Определена система аксиом K_b для BPTL. Доказано, что BPTL полна по отношению ко всем структурам ветвящегося времени.

2. Реализация темпоральной логики ветвящегося времени

С точки зрения практической реализуемости выделяются темпоральные логики, построенные с использованием парадигмы согласования ограничений. Среди них наиболее простой и приемлемой по вычислительной сложности для применения в составе современных интеллектуальных систем типа ИСППР РВ является точечная временная логика [Еремеев и др., 2009]. Эта логика может быть расширена до темпоральной логики ветвящегося времени аналогично расширению логики PLT до BPLT [Куриленко, 2009]. Реализация алгоритмов вывода в точечной временной логике основывается на решении задачи согласования временных ограничений (ЗСВО), являющейся конкретизацией более общей задачи согласования ограничений (ЗСО), что позволяет использовать для решения ЗСВО методы, применяемые для ЗСО [Еремеев и др., 2003, 2005].

ЗСВО задается следующим набором $Z = (V,D,BTR,C)$,

где $V = \{V_1,V_2,\ldots,V_m\}$ – конечное множество временных переменных (моментов времени);

- D – область значений временных переменных (множество целых чисел);
- $BTR=\{r_1,r_2,\ldots r_n\}$ – конечное множество взаимоисключающих бинарных базовых временных ограничений, полное объединение которых является универсальным ограничением U (не накладывающим вообще никаких ограничений);

• $C=\{C_{ij}|C_{ij}=\{r_1,\ldots r_k\};k>0;r_1,\ldots r_k\in BTR;i,j\leq m\}$, конечное множество ограничений, где C_{ij} – ограничение над временными переменными V_i и V_j, интерпретируемое как $(V_i,r_1,V_j)\vee\ldots\vee(V_i,r_k,V_j)$. Если C_{ij} состоит только из одного дизъюнкта, то оно называется *точным*.

Для решения задачи выполнимости ЗСВО **SAT** необходимо найти множество не противоречащих друг другу ограничений $C^*=\{C_{ij}^*|C_{ij}^*=\{r_l\},r_l\ \ C_{ij}\}$. Если такого множества построить нельзя, то ЗСВО является *несогласованной*. Если ЗСВО имеет по крайне мере одно решение, то она называется *согласованной*.

Ограничения между временными примитивами представлены в виде качественных бинарных ограничений, т.к. с их помощью можно представить любые ограничения более высокого порядка.

Основными операциями над временными ограничениями являются:

- отрицание (¬): $\neg L_{ij}=U\backslash L_{ij}$;
- инвертирование (~): $\sim(r_1,\ldots,r_k)=(\sim r_1,\ldots,\sim r_k)$;
- пересечение: $S\cap T=\{r|r\in S,r\in T\}$;
- композиция: $T\bullet S=\{t_1,\ldots t_k\}\bullet\{s_1,\ldots s_q\}=\{t_1\bullet s_1,t_1\bullet s_2,\ldots t_k\bullet s_q\}$.

Известно, что множество всех возможных типов временных ограничений для двух временных примитивов, состоящее из $2^{|BTR|}$ типов ограничений, замкнуто относительно операций ¬, ~, ∩, • и образует *алгебру* временных ограничений [Gereviny et al., 1993].

ЗСВО называют *единичной* ЗСВО (ЕЗСВО), тогда и только тогда, когда в множество C входят только точные ограничения. При этом сама ЗСВО сводится к проверке непротиворечивости (согласованности) ограничений из множества C.

Построим на основе определения ЕЗСВО ЗСВО для ветвящегося времени. Сначала определим на базе ЕЗСВО сценарий в виде $S_i=(V_i,C_i,S_j)$, где $S_0=(V_0,D,BTR,C_0)$ – ЕЗСВО, интерпретируемая как начальный сценарий; S_i – наследуемый сценарий, который расширяет множество переменных V_j и множество единичных ограничений Cj сценария S_j, $j<i$, множествами Vi и Ci соответственно; D – область определения переменных (множество целых чисел), BTR- множество базовых временных ограничений. Для каждого сценария, исходя из определения ЕЗСВО, можно поставить задачи поиска минимального представления и проверки согласованности, т.е. можно говорить о согласованном или несогласованном сценарии и сценарии в минимальном представлении. Будем называть текущими сценариями такие сценарии, для которых не существует наследуемых сценариев.

Определим ветвящуюся ЗСВО как множество альтернативных сценариев, унаследованных от одного начального сценария S_0 $VZ = \{S: S$ – сценарий$\}$. Определим для ветвящейся ЗСВО следующие подзадачи:

- Проверка согласованности – проверка существования как минимум одного текущего согласованного сценария $Si \in VZ$.
- Проверка истинности каких-либо утверждений для конкретного текущего сценария.
- Преобразование всех ЕЗСВО, соответствующих согласованным сценариям, к минимальному виду.
- Проверка истинности каких-либо утверждений для всех текущих сценариев или хотя бы для одного текущего сценария.

Рассмотрим иллюстративный пример (рис. 1). Данная задача содержит 16 сценариев, из которых 5 являются несогласованными. Следует отметить, что, исходя из практических соображений, следует допускать наследование только согласованных сценариев. Поэтому сценарии S^2_6 и S^2_7 не участвовали в порождении сценариев-наследников. Таким образом, каждый сценарий соответствует возможному варианту развития событий (задаваемому соответствующим множеством ограничений).

Известны различные стратегии формирования сценариев в ветвящейся ЗСВО. Источником ветвления на каждом шаге являются дизъюнктивные утверждения. Недизъюнктивные утверждения добавляются к каждому текущему сценарию и не приводят к ветвлению. В случае же дизъюнктивных утверждений к каждому сценарию добавляется ряд наследников. Например, при добавлении в ветвящуюся ЗСВО ограничения типа C_{ij} v C_{xy} для каждого текущего сценария должны быть добавлены три сценария: а) в котором обязательно наличие ограничения C_{ij}; б) в котором обязательно наличие C_{xy}; в) в котором обязательно наличие как C_{ij} так и C_{xy}. Естественно, что при такой постановке ветвление является существенной проблемой, т.к. ведет к сильному росту числа возможных сценариев. Однако существует приемы, которые могут быть использованы для ограничения ветвления. На практике может быть использован вариант внесения ограничений, связанных исключающим *или* (т.е. нет необходимости порождения сценария, в котором обязательно наличие комбинации C_{ij} и C_{xy}). Другой возможной стратегией является независимая достройка сценариев, когда каждый сценарий достраивается и разветвляется так, как это необходимо, исходя из решаемой задачи, и внесение дизъюнктивного ограничения приводит к разветвлению конкретного сценария и может не затронуть все оставшиеся сценарии.

Алгоритмы решения задач ветвящейся ЗСВО могут быть построены на базе алгоритмов решения ЕЗСВО [Gereviny et al., 1993, Еремеев и др., 2005, Куриленко, 2007]. В частности каждую ЕЗСВО в процессе решения ветвящейся ЗСВО можно решать с помощью перехода к задаче на графе времени (TL-графе).

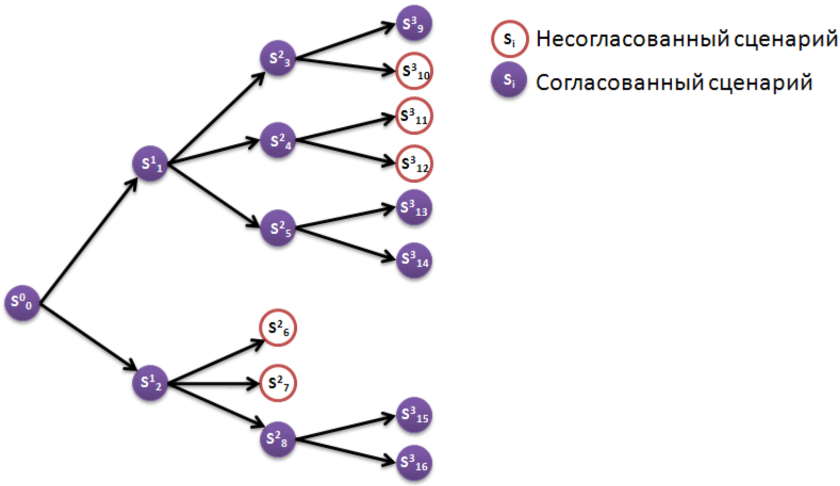

Рис. 1. ЗСВО для ветвящегося времени

В работе [Куриленко, 2009] предлагаются алгоритмы позволяющие экономить вычислительные ресурсы при последовательном дополнении ЕЗСВО ограничениями, за счет анализа производимых изменений. В случае ветвящейся ЗСВО такие алгоритмы хорошо подходят для решения ЕЗСВО для каждого сценария-наследника $S_i=(V_i,C_i,S_j)$, так как они определяются как расширение множества ограничений в сценарии-предке. Данные пошаговые алгоритмы (вычисления выполнимого ограничения, создания и удаления связей в TL-графе) отличаются тем, что позволяют поддерживать в актуальном состоянии множество всех существенных путей на TL-графе, соответствующему ЕЗСВО, после каждого изменения. Преимуществом такого подхода является то, что после каждого изменения не требуется вызов дополнительных алгоритмов проверки согласованности и вычисления неявных ограничений. Данные алгоритмы позволяют организовать эффективное и приемлемое для ИСППР РВ решение ветвящейся ЗСВО и ее подзадач.

Заключение

Рассмотренная темпоральная логика ветвящегося времени BPLT является модальной темпоральной логикой. Главной проблемой при компьютерной реализации такой логики является нахождение полиномиальных подклассов алгоритмов вывода, в общем случае характеризующихся NP- или экспоненциальной сложностью.

Предложенная в работе ветвящаяся логика на базе точечной временной логики является менее выразительной, но достаточно просто реализуемой.

В настоящее время на кафедре прикладной математики МЭИ (ТУ) в плане разработки методов и базовых инструментальных средств конструирования ИСППР РВ на основе нетрадиционных логик разрабатывается система моделирования темпоральных рассуждений, интегрирующая различные модели представления временных зависимостей (для метрического, интервального, смешанного представления времени, линейной и ветвящейся структур времени, количественных и качественных временных зависимостей и т.д.).

Список литературы

[Еремеев, 2006] Еремеев А.П. Логика ветвящегося времени и ее применение в интеллектуальных системах поддержки принятия решений // Сб. тр. 10-й нац. конф. по искусственному интеллекту с междунар. уч. КИИ-2006. В 3-х т. Т.3. М.: Физматлит, 2006. С. 746-754.

[Карпов, 2010] Карпов Ю.Г. MODEL CHECKING. Верификация параллельных и распределенных программных систем. – СПб.: БХВ-Петербург, 2010.

[Torsun, 1998] Torsun I.S. Foundations of Intelligent Knowledge-Based Systems // ACADEMIC PRESS, London, 1998.

[Еремеев и др., 2009] Еремеев А.П., Куриленко И.Е. Применение темпоральных моделей в интеллектуальных системах / Интеллектуальные системы. Колл. монография. Выпуск третий. / Под. Ред. В.М. Курейчика. – М.: Физматлит, 2009, 195 с., 124-139.

[Куриленко, 2009] Курилено И.Е. Система временного вывода для интеллектуальных систем поддержки принятия решений реального времени // Сб. док. междунар. конф. Интегрированные модели и мягкие вычисления в искусственном интеллекте в 2 т. Т.1. – М.:ФизМатЛит, 2009. С 241–252.

[Еремеев и др., 2003] Еремеев А.П., Троицкий В.В. Модели представления временных зависимостей в интеллектуальных системах поддержки принятия решений // Изв. РАН. Теория и системы управления, 2003, № 5. С. 75-88.

[Еремеев и др., 2005] Еремеев А.П., Куриленко И.Е. Реализация временных рассуждений для интеллектуальных систем поддержки принятия решений реального времени // Программные продукты и системы, 2005, № 2. С. 8-16.

[Gereviny et al., 1993] Gereviny A. and Schubert L. Efficient Algorithms for Qualitative Reasoning about Time. Technical report 496, Department of Computer Science, University of Rochester, Rochester, NY, 1993.

[Куриленко, 2007] Куриленко И.Е. О методах улучшения алгоритмов вывода для системы временных рассуждений // Сб. тр. IV–й междунар. научно–практич. конф. Интегрированные модели и мягкие вычисления в искусственном интеллекте в 2 т. Т.1. – М.:ФизМатЛит, 2007. С.149–156.