Фрактальное построение n-мерных гиперкубовых архитектур в структурном пространстве

А.С.Семенов

Аннотация. Рассмотрено фрактальное построение трех классов n-мерных гиперкубовых архитектур в структурном пространстве: правильных гиперкубов, Фибоначчи гиперкубов и гамильтоновых гиперкубов Фибоначчи. Введено определение n-мерных гиперкубов как фрактоидов, которые в качестве схемы построения используют TL-систему. Даны определения соответствующих TL-систем и приведены алгоритмы построения. Построение в структурном пространстве позволяет рассмотреть классы n-мерных гиперкубов в качестве динамических архитектур для самоорганизующегося программного обеспечения. Рассмотрены свойства и определено отношение вложения между классами фрактоидов.

Введение

При построении мультикомпьютерных систем, кроме таких простых архитектур как кольцо, звезда, двухмерная сетка, широко применяются классы n-мерных гиперкубовых архитектур.

Благодаря своим свойствам n-мерный гиперкуб может стать одной из перспективных архитектур для построения самоорганизующегося программного обеспечения (ПО), которое требует высокой устойчивости к ошибкам (толерантности), адаптируемости и эволюции.

В работе введено определение n-мерных гиперкубов как трех классов фрактоидов [1]: правильных гиперкубов, гиперкубов Фибоначчи и гамильтоновых гиперкубов Фибоначчи.

Фрактоид объединяет гиперкубы одного класса, но разной размерности, в самоподобное множество, которое может быть построено динамически. В качестве схемы построения фрактоидов применяется TL-система [2]. Построение в структурном пространстве позволяет рассмотреть классы n-мерных гиперкубов как динамические архитектуры, представленные компонентами и коннекторами. Такой подход применяется при моделировании информационных систем архитектурными языками описания.

Фрактоид правильного n-мерного гиперкуба реализует архитектуру компонент и коннекторов, позволяя эффективно имитировать другие архитектуры. Однако число компонент 2^n фрактоида быстро растет с увеличением n. При увеличении размерности на единицу общее количество компонент удваивается, еще быстрее увеличивается общее количество коннекторов. Это значительно ограничивает выбор количества компонент. Чем больше размер ПО, тем выше его цена и тем больше вероятность ошибки некоторых компонент и/или коннекторов. При возникновении ошибки гиперкуб не работает должным образом.

Фрактоид n-мерного гиперкуба Фибоначчи имитирует многие алгоритмы правильного гиперкуба, но использует меньшее количество коннекторов по сравнению с гиперкубом, в то время как его размер не увеличивается так быстро как у гиперкуба. Он может рассматриваться как производный от фрактоида правильного гиперкуба, у которого вышли из строя некоторые компоненты и система реконфигурировалась. Таким образом могут моделироваться гиперкубовые системы, работающие в деградирующем режиме.

Однако такой гиперкуб не гамильтонов. Гамильтоновы свойства гарантируют, что кольцевая сеть может эмулироваться гиперкубом того же размера. В работе введен фрактоид n-мерного гамильтонова гиперкуба Фибоначчи с кратным числом компонент.

Определено отношение вложения между фрактоидами разных классов гиперкубов и древовидных структур.

Рассмотренные классы фрактоидов позволяют динамически моделировать архитектуры ПО, строить машины произвольных размеров.

1. Построение n-мерных гиперкубов

Следующие понятия: дискретное структурное пространство, фрактоид и фрактальная алгебра [1] используются при построении n-мерных гиперкубов, поэтому приведем их определения.

Определение 1. Будем называть компактное, совершенное и вполне разрывное множество C самоподобным, если существует система преобразования подобия S_i с коэффициентом подобия r_i такая, что $C = S_1(C)$, $S_2(C), \ldots, S_N(C)$, причем $\dim_M(C) = d$, где d - единственное решение уравнения, называемое размерностью подобия, $r_1^d + r_2^d + \ldots + r_N^d = 1$, где r_i – коэффициенты подобия преобразований $S_i(C)$, лежащие в интервале [0,1]. $\dim_M(C)$ - размерность Минковского множества C.

Определение 2. *Дискретное структурное пространство* Ξ на N элементах определяется как множество всех последовательностей

$$p = \bigcup_{i=0}^{\infty} p^i : p^i_1 p^i_2 \ldots p^i_n, \qquad p_n \in \{1, \ldots, N\},$$

где N ≥ i ≥ 1, *p* - образец задан как *p* = <*R, Ct, Cm, M* >,

где *R* = {*is-a, has-a*} - конечное множество правил классификации, где *is-a* – отношение наследование, *has-a* - отношение агрегация;

$Ct = \{ct_1, ct_2, \ldots, ct_n\}$ - множество контейнеров;

$Cm = \{cm_1, cm_2, \ldots, cm_m\}$ - множество компонент;

$M = \{M_{has\text{-}a}, M_{is\text{-}a}\}$ – конечное множество отображений *Cm* и *Ct* для каждого правила из *R*, причем $M_{has\text{-}a}: Cm \rightarrow Ct = \{0, 1\}$,

$M_{is\text{-}a} : Ct \rightarrow Cm = \{0, 1\}$.

Определение 3. *Фрактоид* $F(p, S, e_p)$ - алгебраическая структура,

где *p* – самоподобное множество, образец;

S – система итерированных отображений множества *p* $F: p \rightarrow p$;

e_p – единичный элемент.

Определение 4. Фрактальная алгебра S={Ξ | $\rho_1, \ldots, \rho_N$} – это уникально обратимая СИО (система итерированных операторов) на структурном пространстве Ξ, ρ_i = {<< , +, ÷, ≡ }, соответственно операторы: << - сдвига, + - композиции, ÷ - разбиения, ≡ -прототипирования, N ≥ i ≥ 1 [1,2].

Определение 5. n-мерный гиперкуб Q_n, где n≥1, традиционно определяется как граф, построенный из двух подграфов Q_{n-1} путем соединения корреспондентских вершин. Базовый граф гиперкуба Q_0 нулевой размерности 2^0 состоит из одной вершины.

Одномерный 2^1 гиперкуб Q_1 конструируется соединением двух гиперкубов нулевой размерности. Двухмерный 2^2 гиперкуб Q_2 конструируются из двух одномерных гиперкубов путем соединения корреспонденствующих вершин и состоит из четырех вершин.

В общем n-мерный гиперкуб Q_n конструируется соединением корреспонденствующих вершин в двух (n-1) - мерных гиперкубах.

Определение 6. Построение n-мерного (правильного) гиперкуба Q_n может быть задано шаблоном:

$Q_n = Q_{n-1} + P_{n-1}$,

где Q_{n-1} – гиперкуб, построенный на предыдущем шаге;

P_{n-1} – гиперкуб прототип, получен из гиперкуба Q_{n-1} в результате выполнения оператора прототипирования, т.е. $P_{n-1} \equiv Q_{n-1}$ [1,2].

Рассмотрим построение n-мерных гиперкубов в дискретном структурном пространстве Ξ в виде фрактоида. Определим оператор композиции + для гиперкубов.

Определение 7. n-мерный гиперкуб Q_n, где n≥1, - это фрактоид $F^n_Q(Q_n, S, e_p)$, построенный по схеме S с n шагами итерации из образца e_p. Шаги итерации задают размер фрактоида. Образец e_p - гиперкуб Q_0 нулевой размерности, состоит из одной компоненты, т.е. $e_p = cm_0$, $cm_0 \in Cm$ образца $p \subset \Xi$ (Рис. 1).

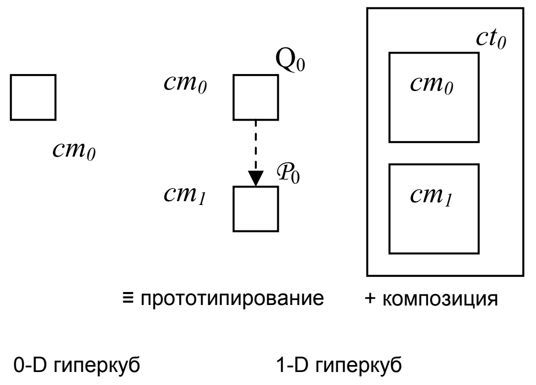


Рис. 1. Операторы: ≡ прототипирования и + композиции n-мерных гиперкубов

Схема S фрактоида F^n_Q - это TL-система, задаваемая шаблоном (Определение 6).

Определение 8. Оператор композиции + прототипа гиперкуба P_{n-1} и Q_{n-1} соединяет коннекторами корреспонденствующие компоненты, результатом является гиперкуб Q_n размерности n.

Например, гиперкуб $Q_0 = cm_0$ прототипируется в $P_0 = cm_1$, (на Рис. 1 штрих пунктирная линия, компоненты изображены квадратиками, коннектор – прямоугольником).

Гиперкуб Q_1 получен композицией корреспонденствующих компонент $Q_1= cm_0 + cm_1$. Контейнер $ct_0 \in Ct$ образца $p \subset \Xi$ используется в качестве коннектора.

Определение 9. Будем называть множество коннекторов β_n, соединяющих корреспонденствующие компоненты фрактоида F^n на шаге n – *n слоем бисекции*, а *шириной бисекции* – мощность слоя. Для F^n_Q гиперкуба *ширина бисекции* $\beta_w = |\beta_n| = \mathbf{2^{n-1}}$ при $n > 0$.

Теорема 1. Количество q_β слоев бисекции β_n фрактоида F^n равно его размерности, $q_\beta = n$.

Доказательство. Схема S фрактоида F^n - это *TL*-система с оператором композиции +. Количество операторов + для F^n равно n. Пусть шаблон $Q_n = Q_{n-1} + P_{n-1}$ соединяет корреспонденствующие компоненты коннекторами. Индуцируя n, образуем первый слой β_1 коннекторов q_β=1, затем следующий слой q_β =2 и т.д., т.е. $q_\beta = n$.

Следствие 1.1. Каждая компонента правильного гиперкуба Q_n связана q_β коннекторами (Рис. 2).

Следствие 1.2. Число коннекторов k правильного гиперкуба Q_n $k= q_\beta \beta_w = n\, 2^{n-1}$.

На Рис. 2 текущий слой бисекции β_n, полученный при композиции, показан штрих-пунктирной линией, коннекторы изображены прямоугольниками, а для 3-мерного и 4-мерного гиперкуба линиями.

При построении n-мерного гиперкуба грамматика TL-системы должна порождать строки *word* длиной *length(word)* = 2^n, где n-размер куба и шаг построения. Обозначим через Q_n строку на шаге *n*, определяемую по шаблону (Определение 6):

$$Q_n = Q_{n-1} + P_{n-1},$$

где образец $P_{n-1} \equiv Q_{n-1}$ получен прототипированием предыдущей строки, + - конкатенация строк. Шаблону соответствуют слова, представленные в Табл. 1.

Символы слова Q_n, порождаемые на каждом шаге *n*, изменяются в зависимости от *n*. Соответственно, в левой части строки от символа +, для Q_{n-1} , а в правой части строки - P_{n-1}. Например, для $n = 3$ Q_n = *GCGC+GCGC,* левая часть Q_{n-1} = *GCGC,* правая часть P_{n-1} = *GCGC.*

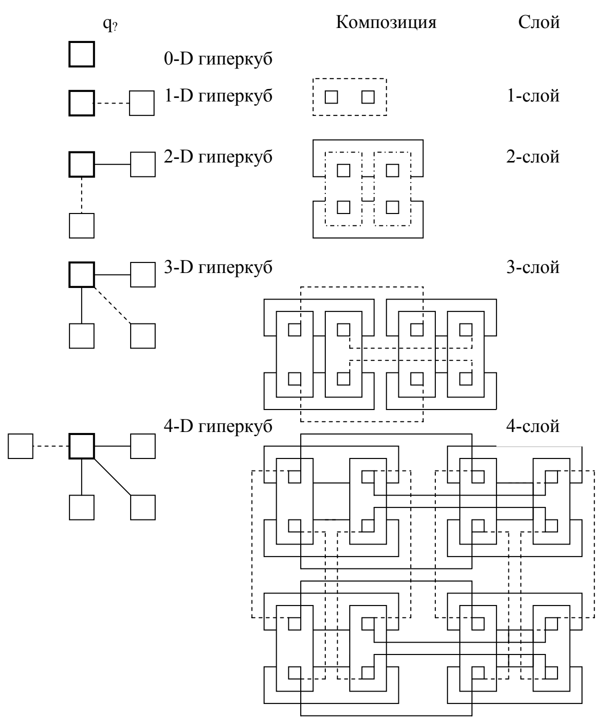


Рис. 2. Модель n-мерных правильных гиперкубов в структурном пространстве

Табл.1.

n	*Строка*	$Q_n = Q_{n-1} + P_{n-1}$	Q_{n-1}	P_{n-1}	$length(Q_n)=2^n$
n_0	*“”*	$axiom_0$ = *“”*	*“”*	*“”*	*0*
n_1	*GC*	*G + C*	*G*	*C*	*2*
n_2	*GCGC*	*GC + GC*	*GC*	*GC*	*4*
n_3	*GCGCGCGC*	*GCGC + GCGC*	*GCGC*	*GCGC*	*8*
n_4	*GCGCGCGCGCGCGCGC*	*GCGCGCGC+GCGCGCGC*	*GCGCGCGC*	*GCGCGCGC*	*16*

Табл. 2.

N	*Строка*	index($Q_{n-1} + P_{n-1}$)	B_n(index(*word*))
n_0	*“”*	*“”*	*0*
n_1	*GC*	$G_0 + C_1$	*0 1*
n_2	*GCGC*	$G_0C_1 + G_2C_3$	*00 01* *10 11*
n_3	*GCGCGCGC*	$G_0C_1G_2C_3 + G_4C_5G_6C_7$	*000 001 010 011* *100 101 110 111*
n_4	*GCGCGCGCGCGCGCGC*	$G_0C_1G_2C_3G_4C_5G_6C_7 +$ $G_8C_9G_{10}C_{11}G_{12}C_{13}G_{14}C_{15}$	*0000 0001 0010 0011* *0100 0101 0110 0111* *1000 1001 1010 1011* *1100 1101 1110 1111*

Языки, порождаемые шаблоном $Q_n = Q_{n-1} + P_{n-1}$ для графа, структуры и строк n-мерного гиперкуба Q_n - эквивалентны.

Определение 10. *TL* -система для схемы S фрактоида $F^n{}_Q(Q_n, S, e_p)$, задаваемая шаблоном $Q_n = Q_{n-1} + P_{n-1}$, имеет следующий вид:

TL = (*V*, *U*, *P*, *A*, *T*) = ({*G* , *C*}, {+,“”}, {*G* ->*GC*, *C* -> *GC*}, {$axiom_0$=“”, $axiom_1$ = *G*, $axiom_2$= *C* }, {*template* = $axiom_1 + axiom_2$})

Теорема 2. Разметка M: B $\rightarrow Q$ n-мерного гиперкуба Q_n, где B ={0,1} – конечное множество, полученная из начальной разметки 0 применением операторов сдвига $<<^n{}_0$ и $<<^n{}_1$, где n – количество бит в кодовом слове, по шаблону

$M(Q_n) = (M(Q_{n-1}) <<^n{}_0) + (M(P_{n-1}) <<^n{}_1)$ - есть код Грея.

Доказательство. Пусть каждому порождаемому символу строки поставлено в соответствие число – порядковый номер символа слева направо, начиная с нуля. Введем функцию индексации символов слова index(*word*) (Табл. 2).

Преобразуем каждый индекс символа слова в бинарное представление функцией B_n(index(*word*)), которая вычисляет бинарное представление каждого индекса символа слова *word*, где n – размерность гиперкуба Q_n соответствует количеству бит в бинарном представлении.

Разметка M n-мерного гиперкуба Q_n M:B $\rightarrow Q$, где B - множество меток, полученных функцией B_n (Табл.2), есть код Грея, так как разметка удовлетворяет правилам:

- число кодовых слов равно 2^n, длина каждого из них составляет n битов,
- соседние кодовые слова различаются только в одном разряде, т.е. расстояние Хемминга между ними равно единице.

Расстояние Хемминга между двумя компонентами – это количество битов на которые отличаются два адреса компонент. Это нижняя граница расстояния между двумя компонентами. Путь расстояния Хемминга - это путь между двумя компонентами с длиной, эквивалентной расстоянию Хемминга этих двух компонент.

Утверждение 2.1. Разметка корреспонденствующих компонент различается только в одном разряде, т.е. расстояние Хемминга между такими компонентами равно единице.

Воспользуемся утверждением 2.1. и введем два оператора сдига:

$<<^n{}_0$ - заполняет бит n нулем,

$<<^n{}_1$ - заполняет бит n единицей, тогда разметка гиперкуба по шаблону $M(Q_n) = (M(Q_{n-1}) <<^n{}_0) + (M(P_{n-1}) <<^n{}_1)$ есть код Грея, где n – количество битов в коде. При прототипировании образца $P_{n-1} \equiv Q_{n-1}$ будем сохранять разметку.

Следствие 2.1. Количество битов в кодовом слове равно размеру гиперкуба, при n >0.

Следствие 2.2. Структура n-мерного правильного гиперкуба Q_n образует гамильтонов цикл, так как размечена кодом Грея и число кодовых слов равно 2^n.

Следствие 2.3. Каждой компоненте n-мерного гиперкуба Q_n соответствует символ слова *word,* порождаемого правилами *P*.

Алгоритм 1. Построение n-мерных правильных гиперкубов Q_n: строки, графа, объектной структуры и разметки гиперкубов кодом Грея.

Для графического построения n-мерного гиперкуба используется система графического вывода, которая прорисовывает по координатам компоненты и контейнеры.

Вход:

$axiom_0$ = “” -- *ставим в соответствие* Q_0

$axiom_1 = G$

$axiom_2 = C$

template = ($axiom_1 + axiom_2$) – *слово, порождаемое правилами грамматики*

P = {*G ->GC, C -> GC* } -- *правила грамматики*

$axiom_3 = cm_0$ -- *компонента (Определение 7)*

Выход:

Q_n -- *размеченная* объектная структура n-мерного гиперкуба и его графическое представление

Инициализация:

level=4 - размерность гиперкуба*, который надо построить*

word = template

n = *0*

$Q_n = axiom_3$ *- Определение 6:* гиперкуб n = *0, одна компонента*

Шаги:

for *i=1* to *level* - построить гиперкуб заданной размерности *level*

n = i

$P_{n-1} \equiv Q_{n-1}$ - прототипировать гиперкуб Q_{n-1} в образец P_{n-1}

$M(Q_n) = (M(Q_{n-1})<<^n{}_0) + (M(P_{n-1})<<^n{}_1)$ -- *разметить и соединить* корреспонденствующие компоненты

rule (word) - сгенерировать следующее слово (Табл. 1).

end for

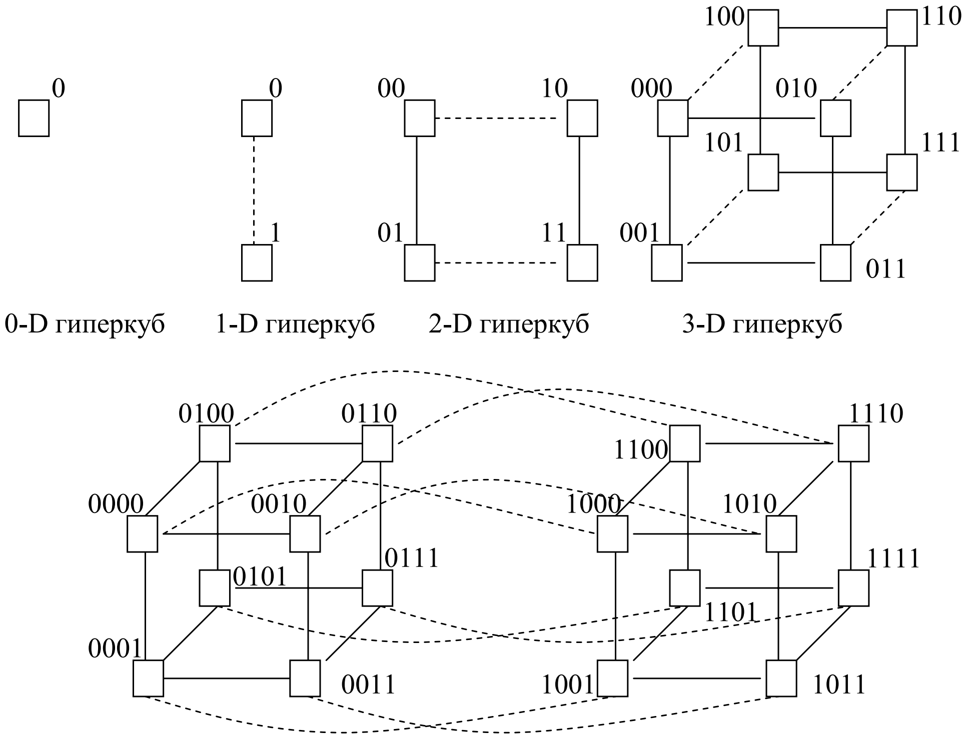
Графы, соответствующие шагам построения объектной структуры 4-мерного правильного гиперкуба, проиллюстрированы на Рис. 2. На Рис. 3. дано другое графическое изображение, соответствующее шагам построения 4-мерного гиперкуба, помеченного кодом Грея.

2. Построение n-мерных гиперкубов Фибоначчи

Определение 11. n-мерный гиперкуб Фибоначчи Q_n, где n≥0, - это фрактоид $F^n{}_F(Q_n, S, e_f)$, построенный по схеме S на основе ряда Фибоначчи {1,2,3,5,8,13,….}, с n шагами итерации из образца e_f. На каждому шаге n строится гиперкуб Q_n. Количество компонент гиперкуба f_n равно числу Фибоначчи n. Образец e_f - гиперкуб Q_0 нулевой размерности, n=0, состоит из одной компоненты, т.е. $e_f = cm_0$, $cm_0 \in Cm$ образца $p \subset \Xi$ (Рис. 1). Ряд Фибоначчи {1,2,3,5,8,13,….} выбран для соответствия количества слоев бисекции β_n с правильным гиперкубом. В классике ряд Фибоначчи {0,1,1,2,3,5,8,13,….}.

Схема S фрактоида $F^n{}_F$ - это *TL*-система с шаблоном $P_n = P_{n-1} + P_{n-2}$ (подробно изложена для построения дерева Фибоначчи в [2]). С учетом структурного пространства система определена следующим образом: *TL*=(*V*, *U*, *P*, *A*, *T*)= ({*C*,*G*}, {+,“”}, {*C->G*, *G->GC*}, {$axiom_0$="", $axiom_1$=*G*, $axiom_2$=*C*, $axiom_3$=cm_0},{*template*=$axiom_1$+$axiom_2$})

Каждый образец именуется порождаемым словом, например, если строка *GCG*, то образец имеет имя P_{GCG} . Оператор *L(word)* возвращает значение слова слева от терминального символа +, *R(word)* - справа. Например, *word = GC+G, L(word) = GC*, *R(word) = G*. Образцы хранятся в стеке по именам – *Stack {word,* P_{word}*}*.



4-D гиперкуб

Рис. 3. Построение 4-мерного правильного гиперкуба

Алгоритм 2. Построение n-мерного гиперкуба Фибоначчи Q_n : строки, графа, объектной структуры и разметки гиперкубов кодом Грея.

Вход:

$axiom_0$ = "" - ставим в соответствие Q_0

$axiom_1 = G$ - на первом шаге символу ставится в соответствие образец G

$axiom_2 = C$

template = $axiom_1 + axiom_2$ - слово порождаемое правилами грамматики

P = {C -> G , G ->GC} - правила грамматики

$axiom_3 = cm_0$

Выход:

Q_n -- маркированная кодом Грея объектная структура n-мерного гиперкуба Фибоначчи и его граф

Инициализация:

level = 4 - размер гиперкуба

word = template

n = 0

$Q_n = axiom_3$

Stack {$axiom_1$,Q_n} -- первый элемент в стеке компонента cm_0 под именем G

Stack {$axiom_2$,Q_n} -- второй элемент в стеке компонента cm_0 под именем C

Шаги:

for *i =1* to *level* -- построить гиперкуб заданной размерности *level*

n = i - размерность гиперкуба в зависимости от шагов

P_{n-2} = Stack { R(word), P_{word} } - получить предыдущий образец

$M(Q_n) = (M(Q_{n-1}) \ll^{n}{}_{0}) + (M(P_{n-2}) \ll^{n}{}_{1})$ - разметить и соединить

rule (word) - сгенерировать следующее слово по грамматике

$P_n \equiv Q_n$ - прототипировать Q_n в образец P_n

Stack { L(word), P_n } -- сохранить образец в записи по имени

end for

На Рис. 4. показано построение маркированного кодом Грея 4-мерного гиперкуба Фибоначчи и его графа.

При разметке F-куба и построении графа используется Теорема 2.

Теорема 3. *TL*-системы подобны, если основаны на одинаковых числовых рядах. Алгоритмы построения архитектур отличаются реализацией операторов фрактальной алгебры.

Доказательство. *TL*-система гиперкуба Фибонначи подобна *TL*-системе дерева Фибонначи (Ф-дерева) [2]. Ф-дерево является остовным деревом для гиперкуба Фибонначи. *TL*-системы подобны, алгоритмы построения отличаются реализацией операторов композиции и начальными значениями.

TL-система правильного гиперкуба отличается от *TL*-системы бинарного дерева реализацией оператора композиции. Бинарное дерево является остовным деревом для правильного гиперкуба.

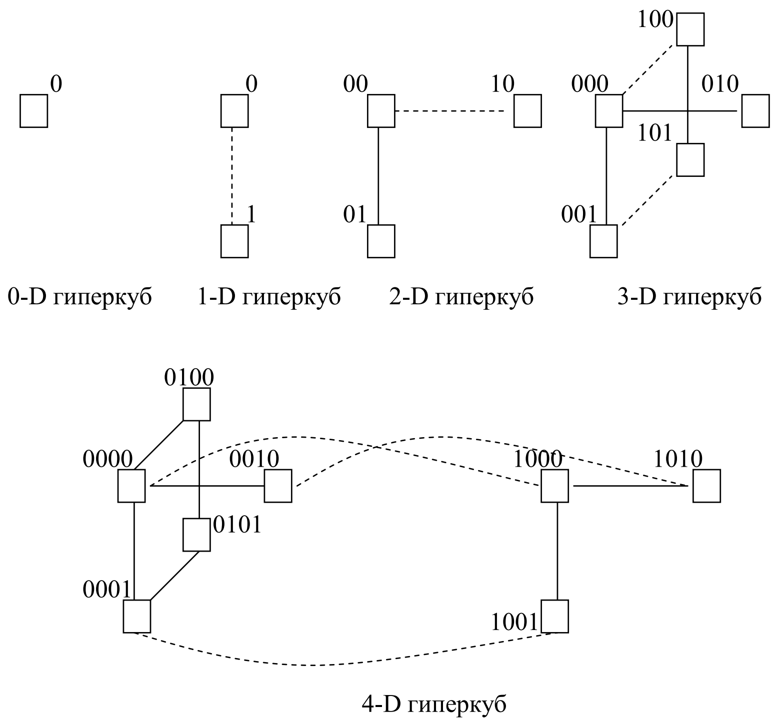


Рис. 4. Построение 4-мерного гиперкуба Фибоначчи в структурном пространстве

Теорема 4. Фрактоиды F^n, основанные на числовых рядах с нечетными числами, строят не гамильтоновы гиперкубы.

Доказательство. Кодовые слова корреспонденствующих компонент F^n отличаются только одним разрядом, т.е. расстояние Хемминга равно единице. Однако число кодовых слов будет нечетным, если в гиперкубе нечетное количество компонент, т.е. не равно 2^n (Теорема 2, Следствие 2.3).

Следствие 4.1. Гиперкубы $F^n{}_F$ не гамильтоновы.

3. Построение n-мерных гамильтоновых гиперкубов Фибоначчи

Фрактоид n-мерного гамильтонова гиперкуба Фибоначчи с кратным числом компонент строится подобно фрактоиду $F^n{}_F$ (Q_n, S, e_f), но его начальные значения инициализации - всегда четное количество компонент. Гамильтоновы гиперкубы Фибоначчи исследовались в работах [3,4], в данной работе фрактоид обобщает этот класс, что упрощает его анализ и применение. Фрактоид поддерживает все свойства гиперкубов Фибоначчи, которые являются гамильтоновыми. Такое представление дает больше возможностей для построения систем различных размеров и улучшает имитацию правильных гиперкубов.

Определение 12. n-мерный гамильтонов гиперкуб Фибоначчи Q_n, где n≥0, - это фрактоид $F^n{}_{GF}$ (Q_n, S, e_{gf}), построенный по схеме S на основе ряда Фибоначчи {2,4,6,10,16,….}, с n шагами итерации из образца e_{gf}. На каждому шаге n строится гиперкуб Q_n, количество компонент в гиперкубе соответствует числу в указанном ряде Фибоначчи. Образец e_{gf} - гиперкуб Q_1 первой размерности, n=1, состоит из двух компонент, т.е. $e_{gf} = cm_0 + cm_1$, где $cm_0, cm_1 \in Cm$ образца $p \subset \Xi$ (Рис. 5).

Шаблон для гамильтонова гиперкуба Фибоначчи с четным количеством компонент $f_0=2$, $f_1=4$, где f_n число Фибоначчи n, имеет вид $P_n = P_{n-1} + P_{n-2}$, *TL*-система определяется следующим образом *TL*= (*V*, *U*, *P*, *A*, *T*) = ({*C*, *G*}, {+, “”}, {*C->G*, *G ->GC*}, {*axiom*$_0$ = "", *axiom*$_1$ = *GG*, *axiom*$_2$= *CC*, *axiom*$_3$ = $cm_0 + cm_1$}, {*template*= *axiom*$_1$ + *axiom*$_2$})

Теорема 5. Фрактоиды F^n с подобными *TL*-системами и операторами композиции имеют эквивалентные алгоритмы построения.

F^n_F подобен F^n_{GF} (Теорема 3), *TL*-системы подобны, тогда алгоритмы построения эквивалентны.

<u>**Алгоритм 3.**</u> Построение n-мерного гамильтонова гиперкуба Фибоначчи Q_n: строки, графа, объектной структуры и разметки гиперкубов кодом Грея.

<u>Вход:</u>

$axiom_0$ = "" - *ставится в соответствие* Q_0

$axiom_1 = GG$

$axiom_2 = CC$

$template = axiom_1 + axiom_2$ - *слово порождаемое правилами грамматики*

$P = \{C \ \text{->}\ G\ ,\ G\ \text{->}GC\ \}$ - *правила грамматики*

$axiom_3 = cm_0 + cm_1$ - гиперкуб Q_1 первой размерности

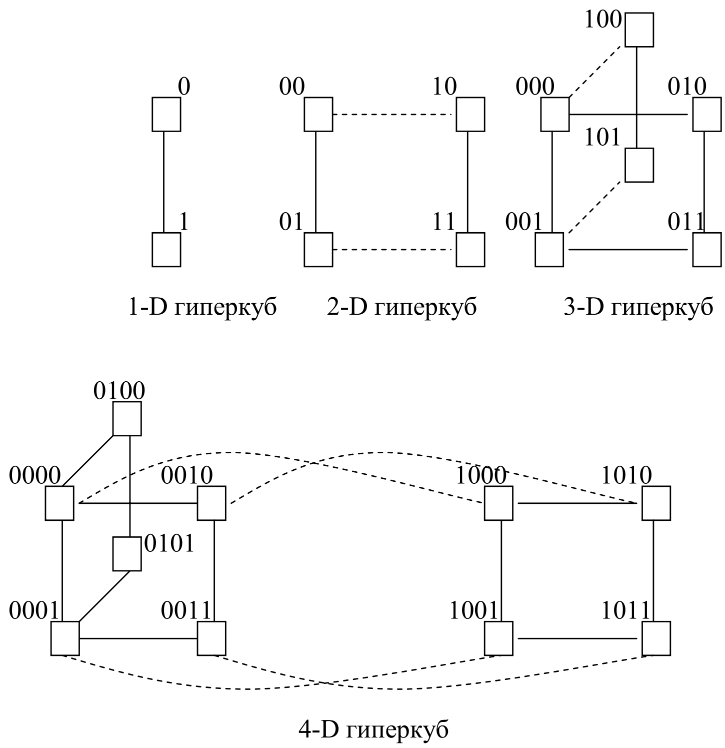


Рис. 5. Построение 4-мерного гамильтонова гиперкуба Фибоначчи

<u>Выход:</u>

Q_n - маркированная кодом Грея объектная структура n-мерного гамильтонова гиперкуба Фибоначчи и его граф

<u>Инициализация:</u>

$level = 4$ - *размер гиперкуба*

$word = template$

$Q_1 = axiom_3$

$Stack\ \{axiom_1,\ Q_1\}$ - *первый элемент в стеке* Q_1 *под именем GG*

$Stack\ \{axiom_2, Q_1\}$ - *второй элемент в стеке* Q_1 *под именем CC*

<u>Шаги:</u>

for $i = 2$ to *level* - построить гиперкуб заданной размерности $level > 1$

$n = i$ - *размерность гиперкуба в зависимости от шагов*

$P_{n-2} = Stack\ \{\ R(word),\ P_{word}\ \}$ - *получить предыдущий образец*

$M(Q_n) = (M(Q_{n-1}) \ll^{n}{}_{0}) + (M(P_{n-2}) \ll^{n}{}_{1})$ - *разметить и соединить*

rule (word) - *сгенерировать следующее слово по грамматике*

$P_n \equiv Q_n$ - прототипировать Q_n в образец P_n

$Stack\ \{\ L(word),\ P_n\ \}$ - сохранить образец *в записи по имени*

end for

На Рис. 5. показано построение маркированного кодом Грея 4-мерного гамильтонова гиперкуба Фибоначчи и его графа.

Теорема 6. Ширина слоя бисекции фрактоидов Фибонначи $\beta_w = f_{n-2}$. Доказательство следует из определения оператора композиции + и шаблона *TL*-системы.

4. Сравнительный анализ исследованных классов фрактоидов

Свойства и отношение между фрактоидами важны для моделирования ПО.

Определение 13. Степень разреженности $Grid = F^n_A/F^n_B$ фрактоида F^n_A - это отношение его числа компонент к числу компонент объемлющего фрактоида этого же размера n. Отношение отражает использование компонент в объемлющих кубах.

Теорема 7. Степень разреженности $Grid = F_A/F_B$ фрактоида F^n_A инвариантна, т.е. не зависит от размера n.

Табл. 3. Свойства фрактоидов

Фрактоид	Число компонент	*Grid* - степень разреженности	Число коннекторов k	Число коннекторов компоненты
F_{TB} - бинарное дерево	$2^h - 1$	(7/8)=0,875 (h=3)	$2^h - 2$	$cm^1_i=1; cm^h_i=2; cm^{h-1}_i=3, h>1$
F_Q - правильный гиперкуб	2^n	(8/8)=1	$n\,(2^{n-1})$	$cm^n_i = n = q_\beta$
F_{TF} - Фибоначчи дерево	$f_{h+1} - 1$	(4/8) =0,5 (h=3)	$f_{h+1} - 2$	$cm^1_i=1; cm^h_i=2; cm^{h-1}_i=3, h>1$
F_F - Фибоначчи гиперкуб	f_n	(5/8) = 0,625	$(2(n+1)f_n - (n+2)f_{n-1})/5$, $n>1$	$n-2 \le cm^n_i \le n$
F_{GF} - Фибоначчи гамильтонов гиперкуб	f_n, при $f_n \bmod 2=0$	(6/8) = 0,75	$k_n = k_{n-1} + f_{n-2} + k_{n-2}$, где $k_1=1$, $k_2=4$, $n>2$	$n-2 \le cm^n_i \le n-1$

Доказательство. Пусть $F^n_B = F^n_Q$, а F^n_A - любой фрактоид из Табл.3, например F^n_F. Индуцируя n F^n_F и F^n_Q, получим, что Grid = F_F/F_Q константа, F_Q - фрактоид, взятый за основу Grid $(F_Q) = 1$.

В Табл.3 приведены отношения Grid при n=3 и высоте h=3 полностью сбалансированного бинарного дерева и Ф-дерева [2], f_n и f_h – числа Фибоначчи.

Определение 14. Отношение вложения $F_A \subset F_B$ определено между классами фрактоидов, если степень разреженности Grid (F_A) < Grid(F_B) и k - число коннекторов компоненты cm^k_i фрактоидов $k_A \le k_B$.

$$F_Q \subset F_{GF} \subset F_F, F_Q \subset F_{TB} \subset F_{TF}, F_{GF} \subset F_F \subset F_{TF}$$

Коэффициент Grid определяет порядок вложения, естественно, чем больше Grid, тем лучше вложение.

Заключение

Введены определения трех классов фрактоидов гиперкубов. Исследованы их свойства и отношения между ними.

Грамматики TL-систем задают множество всех возможных конфигураций компонент в ПО. При проведении изменений, пользуясь фрактоидами, можно выявить инвариант для всех возможных архитектур, не нарушающий общие правила построения.

Фрактоиды могут быть использованы для моделирования эволюции самоорганизующегося ПО, описываемого компонентами и коннекторами.

Литература

1. А.С. Семенов. Фрактальная алгебра как основа фрактальной парадигмы программирования // Информационные технологии и Вычислительные системы. – М.: №2, 2006.
2. А.С. Семенов. Построение класса фрактальных систем по шаблону на примере дерева Фибоначчи // Информационные технологии и Вычислительные системы. – М.: №2, 2005 стр.10-17.
3. W. J. Hsu. Fibonacci cubes: a new interconnection topology. IEEE Transactions on Parallel and Distributed Systems. 4, (1), Jan. 1993, 3-12.
4. J. Wu. Extended Fibonacci cubes. IEEE Transactions on Parallel and Distributed Systems. 8, (12), Dec. 1997, 1203-1210.

Семенов Александр Сергеевич. Родился в 1959 г. Окончил МАИ в 1982 году. Докторант МАИ, кандидат физико-математических наук, доцент. Область интересов – развитие общей и прикладной теории фрактальных архитектур информационных и вычислительных систем.