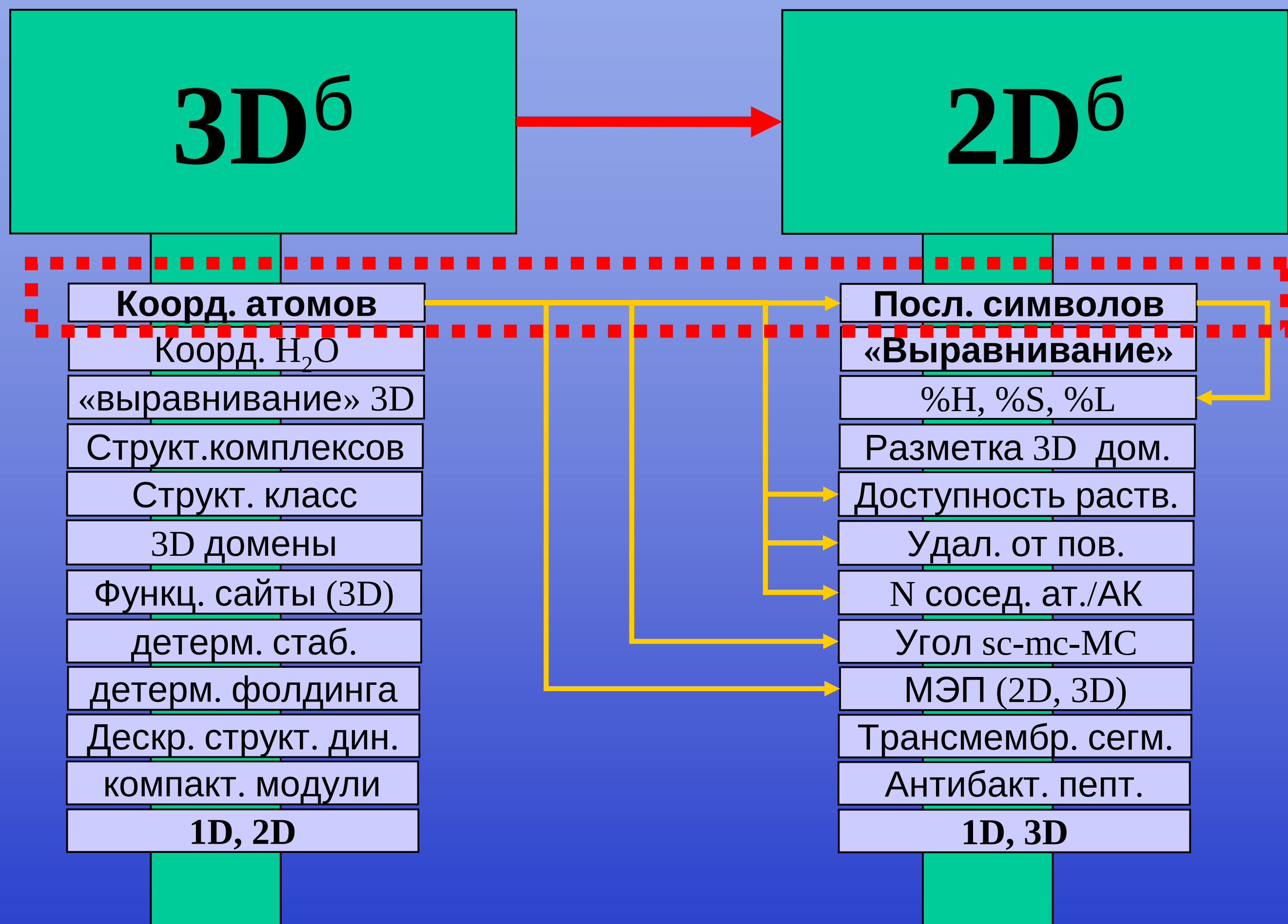
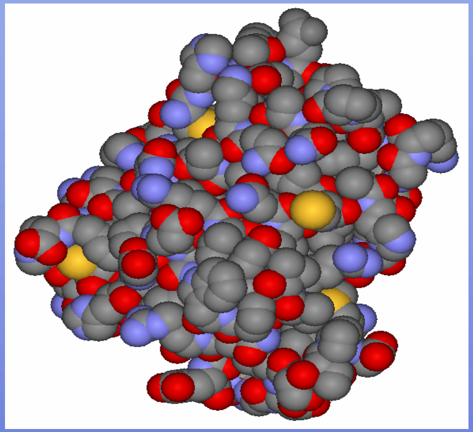
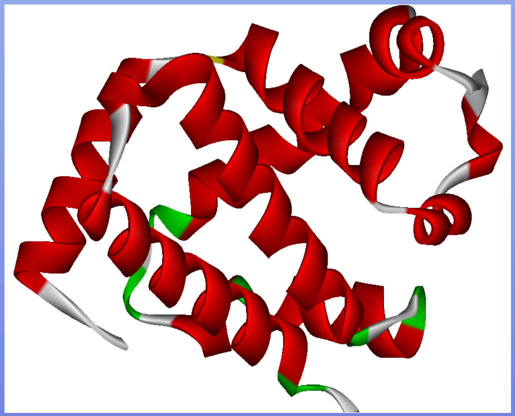
Задачи 3D → 2D



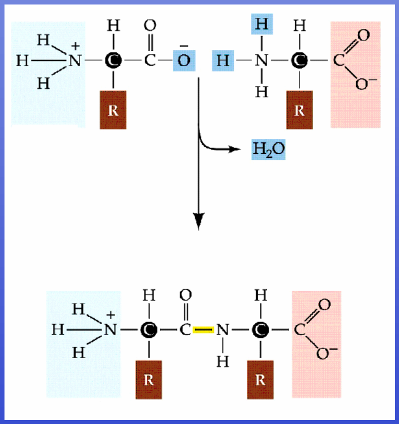
- **Задачи 3D**→2D как «развертки» 3D структуры белка

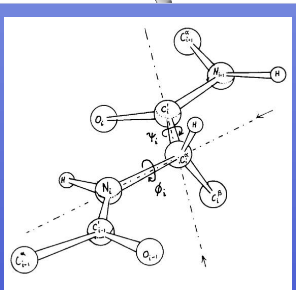
Распознавание 2D в 3D

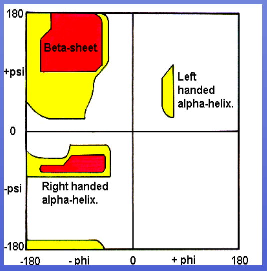


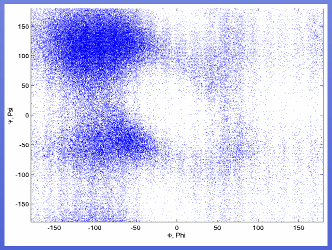


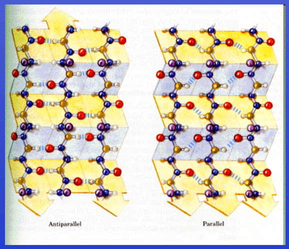
LLLHHHHHHHHHLLLLLL
LHHHHHHHHHHHHLLLLLL
LLLLLLLLLLLLLLLLLLL
LLLLLLLLLLHHHHHHHH
HHHHHLLLLLLLLLHHH
HHHHHHLLLLLLLLHH
HHHHHHHHHLLLLLLLL
LLLLLHHHHHHHHHHHHHH
HLLLLLLLLLLLLLLLL

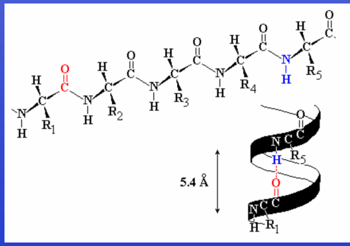




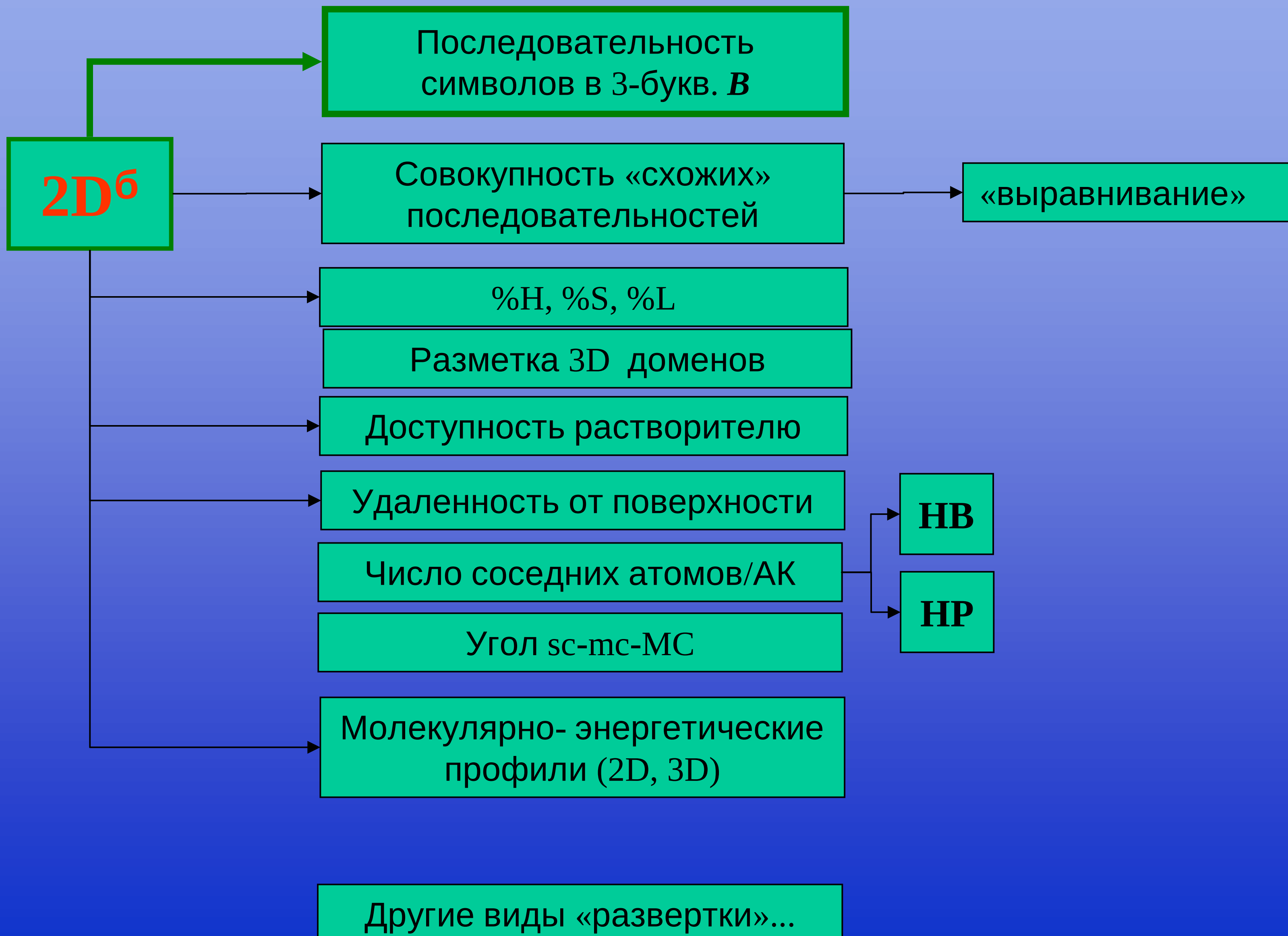








Белок – 2D


2D⁶
Последовательность символов в 3-букв. *B*
Совокупность «схожих» последовательностей
«выравнивание»
%H, %S, %L
Разметка 3D доменов
Доступность растворителю
Удаленность от поверхности
Число соседних атомов/АК
Угол sc-mc-MC
HB
HP
Молекулярно- энергетические профили (2D, 3D)
Другие виды «развертки»...

Структура предположений стоящих за методами в «биоинформатике»

- **Исходные данные**
  - Структурный эксперимент дает однозначную информацию о вторичной структуре данного белка
- **Материал обучения**
  - «Близкие» аминокислотные последовательности имеют «практически идентичную» вторичную структуру
- **Построение алгоритма**
  - Достаточно небольшого числа выравниваний в обучающей выборке (для избежания переобучения)
- **Воплощение алгоритма**
  - Нейронные сети, опорные машины векторов и «скрытые цепи Маркова» - самые лучшие методы информатики, применимые без изменений и к данной задаче
- **Оценка эффективности алгоритмов**
  - Контрольная выборка должна быть достаточно малой (так как обучающая выборка мала)
  - Эффективность к. л. метода всегда повышается за счет **усреднения** результатов, произведенных аналогичными методами

Задача о распознавании вторичной структуры белка

- распознавание = *перевод последовательности из 20-буквенного алфавита в 3-буквенный*

$\vec{a}$

VHLTPEEKSA...

LLLHHHHHHH...

$\vec{b}$

Алфавит «А»

$A^* = \bigcup_{l=1}^{\infty} A^l$

A={A,C,D,E,F,G,H,I,K,L,M,N,P,Q,R,S,T,V,W,Y}

Алфавит «B»

$B^* = \bigcup_{l=1}^{\infty} B^l$

B={**H**,**S**,**L**}

{ VHLTPEEKSAVTALWGKVNVDEVGGEALGRLLVVYPWTQRFFESFGDLSTPDAVMGNPKVKAHGKKVLGAFSDGLAHLDNLKGTFATLSELHCDKLHVDPENFRLLGNVLVCVLAHHFGKEFTPPVQAAYQKVVAGVANALAHKYH

LLLHHHHHHHHLLLLLLLHHHHHHHHHHLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLHHHHHHHHHHHHLLLLLLLLLHHHHHHHHHLLLLLLLLLHHHHHHHHHHHLLLLLLLLLLLLLHHHHHHHHHHHHHLLLLLLLLLLLLLLL }

Решение - функция $\boldsymbol{F}: A^* \rightarrow B^*, \; |F(\vec{a})| = |\vec{a}|$

Разрешимость задачи распознавания вторичной структуры

- Задача *Z(Pr)* разрешима на множестве прецедентов *Pr*, если любой первичной структуре можно однозначно сопоставить некую вторичную структуру

Алфавиты: *A, B*

A={A,C,D,E,F,G,H,I,K,L,M,N,P,Q,R,S,T,V,W,Y,Δ}

B={H,S,L,Δ}

Множества слов: *A*, B*

$$A^* = \bigcup_{l=1}^{\infty} A^l \qquad B^* = \bigcup_{l=1}^{\infty} B^l$$

Множество прецедентов: *Pr*

$$Pr \subseteq A^* \times B^*, \; Pr \neq \varnothing$$

Th.1

Корректность F

$$\forall_{Pr} (\vec{a}, \vec{b}) : F(\vec{a}) = \vec{b}$$

Разрешимость Z

$$\forall_{Pr} (\vec{a}_1, \vec{b}_1), (\vec{a}_2, \vec{b}_2) : (\vec{a}_1 = \vec{a}_2) \Rightarrow (\vec{b}_1 = \vec{b}_2)$$

Регулярность Z

$$\forall_{Pr} (\vec{a}_i, \vec{b}_i), (\vec{a}_j, \vec{b}_j), (i \neq j) \Rightarrow (\vec{a}_i \neq \vec{a}_j)$$

Производные алфавиты на основе последовательностей литер В

- N символов в ***A***-алфавите → $N - 1$ пар базового ***B***-алфавита.
- Пример: B={*H*, *S*, *L*}
  - *B' = {HH, HS, HL, SS, SH, SL, LL, LH, LS}.*

VHLTPEEKSA...

↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓

LLLHHHHHHH...

HH	***HS***	***HL***	***SS***	***SH***	***SL***	***LL***	***LH***	***LS***
0.33	0.001	0.03	0.19	0.003	0.05	0.32	0.03	0.05

Расширение базового алфавита с учетом сегментной структуры слов B

- базовый B-алфавит из m литер трансформируется в $(m + 2\cdot m\cdot(m - 1))$-буквенный
  - $2\cdot m\cdot(m - 1)$ границ сегментов

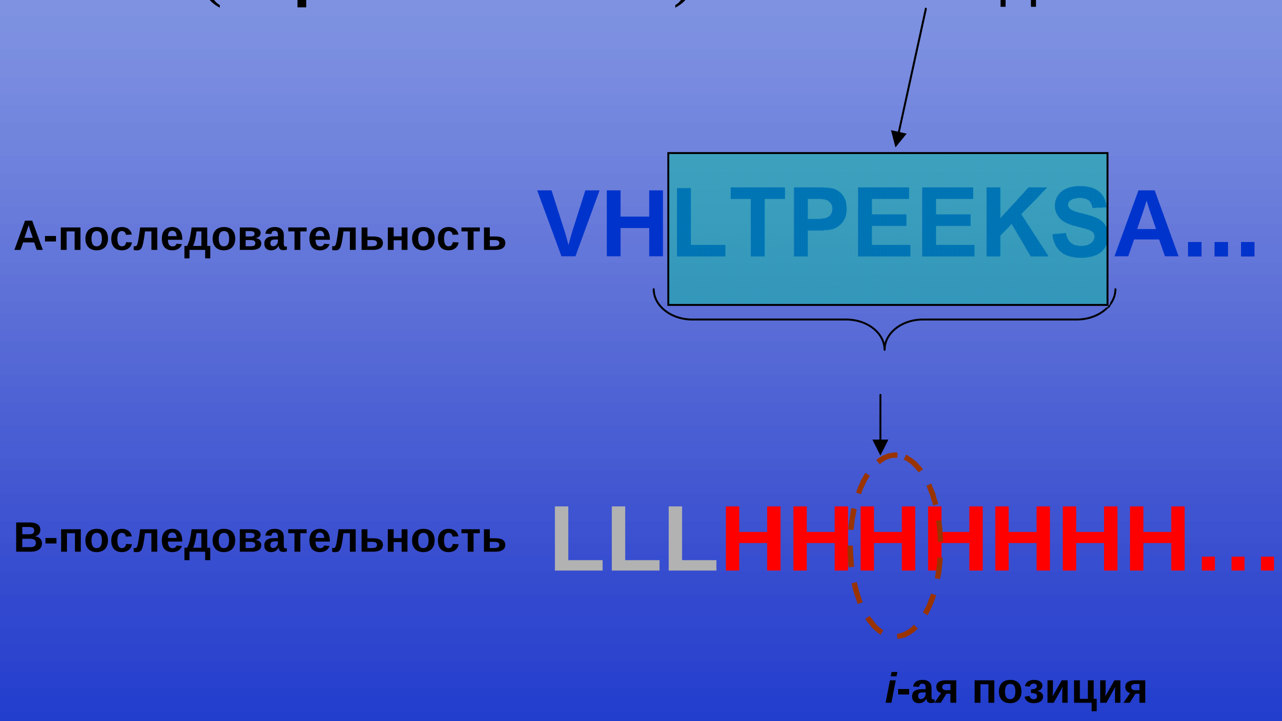
- Пример: 3-буквенный → 15-букв. расширенный

$$\ldots LLL_H^e H_L^b HH \ldots$$

H	S	L	H_S^b	H_L^b	H_S^e	H_L^e	S_H^b	S_L^b	S_H^e	S_L^e	L_H^b	L_S^b	L_H^e	L_S^e
0.33	**0.19**	**0.32**	**0.0015**	**0.015**	**0.0005**	**0.015**	**0.0005**	**0.025**	**0.0015**	**0.025**	**0.015**	**0.025**	**0.015**	**0.025**

Гипотеза: возможна локальная разрешимость задачи

- В *i*-ой позиции, состояние В-последовательности определяется локальным контекстом (**окрестностью**) в А-последовательности



Формальное описание локальности

слово $\vec{U} = \{u_1, u_2, ..., u_n\}$

ведущая позиция i, $1 \le i \le n$

«маска» $\hat{m} = \{\mu_1, \mu_2, \ldots, \mu_m\}$

позиции маски $\mu_i \in Z$

$\mu_1 < \mu_2 < \ldots < \mu_m$

размерность *маски*

$|\hat{m}| = m$

протяженность *маски*

$[\hat{m}] = \mu_m - \mu_1 + 1$

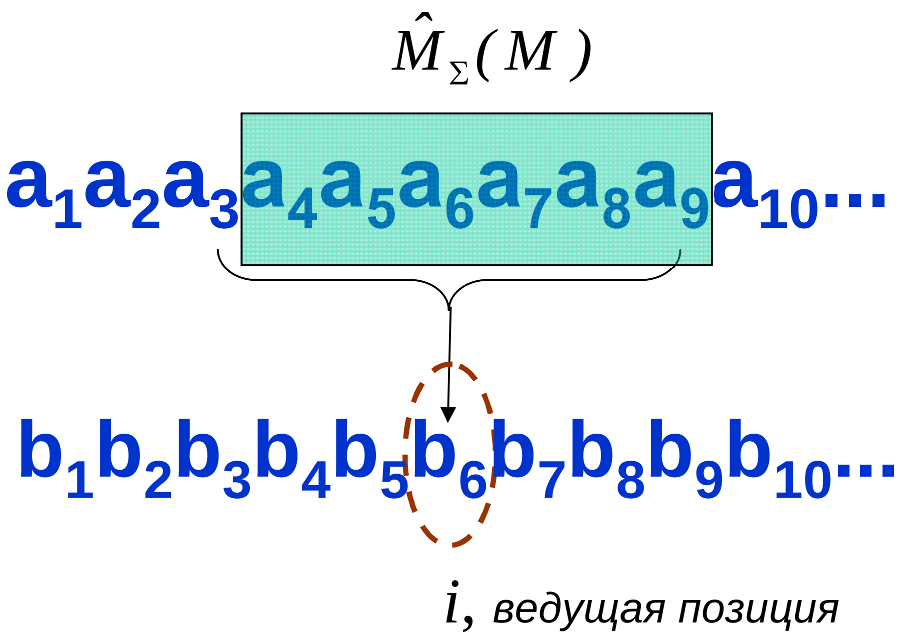
оператор выбора подслова $\eta(i, \hat{m}, \vec{U})$

$$\eta(i, \hat{m}, \vec{U}) = \begin{cases} u_{i+\mu_1} u_{i+\mu_2} \ldots u_{i+\mu_m}, & \text{если } i + \mu_1 \ge 1, i + \mu_m \le n, \\ \varnothing & \text{в противном случае.} \end{cases}$$

система масок $M = \{\hat{m}_1, \hat{m}_2, \ldots, \hat{m}_N\}$

объединенная маска

$$M_\Sigma(M) = \bigcup_{k=1}^{|M|} \hat{m}_k$$



Критерий локальной разрешимости с использованием отдельных масок

Критерий локальной разрешимости

(1) $$\underset{Pr}{\forall}(\vec{V}_1,\vec{W}_1),(\vec{V}_2,\vec{W}_2)\underset{i,j\in N}{\forall}(i,j):\eta(i,\hat{M}_\Sigma(M),\vec{V}_1)=\eta(j,\hat{M}_\Sigma(M),\vec{V}_2)\Rightarrow W_1^i=W_2^j$$

с использованием отдельных масок:

(1”) $$\underset{Pr}{\forall}(\vec{V}_1,\vec{W}_1),(\vec{V}_2,\vec{W}_2)\forall(i,j)\left(\underset{k=1}{\overset{|M|}{\forall}}\hat{m}_k:\eta(i,\hat{m}_k,\vec{V}_1)=\eta(j,\hat{m}_k,\vec{V}_2)\right)\Rightarrow W_1^i=W_2^j$$

$$l(M)<i\leq|\vec{V}_1|-r(M)\quad l(M)<j\leq|\vec{V}_2|-r(M),\ i\neq j$$

Th 4. *Условия (1) и (1”) эквивалентны*

0-тупиковость и тупиковость систем масок

- *M*, при которой задача разрешима, может быть избыточна (разрешимость сохранится при удалении некоторых масок)

- Поиск безызбыточных систем масок должен проводиться среди множества подмножеств объединённой маски $M_\Sigma(M)$ при *L*, *R* = *const*.

0-тупиковая M

условие (1”) выполнено для M, но не выполнено $\forall M' \subset M \;\; M_\Sigma(M') \subset M_\Sigma(M)$

тупиковая M

условие (1”) выполнено для M, но не выполнено $\forall M' \subset M$

Тупиковость и ядерность систем/подсистем масок

Дана 0-тупиковая система масок М

$$\{\hat{m}_1, \hat{m}_2, \ldots, \hat{m}_N\}$$

Ядерная маска $\hat{m}_{i_0}$ $i_0 \in \{1..N\}$

$$\hat{m}_{i_0} \not\subset \bigcup_{j=1,N}^{j \neq i_0} \hat{m}_j$$

Свойство ядерности **систем и подсистем масок**

$$(2) \quad \mathop{\forall}_{i=1}^{N} i \; \mathop{\exists}_{\hat{m}_i} \mu : \mathop{\forall}_{j=1}^{N, i \neq j} (\mu \notin \hat{m}_j)$$

Теорема 5. *М – тупиковая система масок тогда и только тогда, когда М обладает свойством ядерности.*

Необходимость: (2') $\forall i \in \{1..N\}$ $\hat{m}_i \not\subset \bigcup_{j=1,N}^{j \neq i} \hat{m}_j$

Допустим, что (2') нарушено $\hat{m}_i \subseteq \bigcup_{j=1,N}^{j \neq i} \hat{m}_j$

$$\bigcup_{i=1,N} \hat{m}_i = \bigcup_{j=1,N}^{j \neq i} \hat{m}_j \rightarrow M_\Sigma(M') = M_\Sigma(M)$$

т.е. (1") выполнено для $M' = \{\hat{m}_1, \hat{m}_2, \ldots, \hat{m}_{i-1}, \hat{m}_{i+1}, \ldots, \hat{m}_N\}$

⇒ М – не тупиковая

Достаточность:

$$(2) \rightarrow M_\Sigma(M') \subset M_\Sigma(M)$$

Сл.1. Из тупиковости следует 0-тупиковость.

Сл.2. Если в некоторой 0-тупиковой системе масок М имеется ядерная маска , то она входит во все тупиковые подсистемы М

Сл. 3. Пусть в 0-тупиковой М есть несколько ядерных масок $\hat{m}_{i_1}, \hat{m}_{i_2}, \ldots, \hat{m}_{i_L}$ *(ядерная подсистема). Если некоторая* $\hat{m} \subseteq \bigcup_{j=1,L} \hat{m}_{i_j}$ *то она не входит ни в одну тупиковую М.*

Алгоритм построения безызбыточных M

$$\textbf{(1)}\quad \forall_{Pr}(\vec{V}_1,\vec{W}_1),(\vec{V}_2,\vec{W}_2)\forall_{i,j\in N}(i,j): \eta(i,\hat{M}_\Sigma(M),\vec{V}_1)=\eta(j,\hat{M}_\Sigma(M),\vec{V}_2)\Rightarrow W_1^i=W_2^j$$