MTA SZTAKI Közlemények 33/1985 p. 69-85

О ЛИНЕЙНОЙ СЛОЖНОСТИ РЕАЛИЗАЦИИ НЕКОТОРЫХ ОПЕРАТОРОВ СДВИГА

Р.Л. Шчепанович

Математический институт,
Университет Новы Сад, г. Новы Сад,
Югославия

Будем рассматривать реализацию одного класса операторов в классе схем из функциональных элементов в базисе &, V, - /определение см., например, в [1]/. Сложность L(S) схемы S определим как число элементов в ней. Остальные функции Шеннона определим как обычно: сложность L(F) оператора F, это наименьшая из сложностей схемы, реализующих F, и наконец, сложность $L(\mathcal{F})$ класса операторов $\mathcal{F}$ - это $\max\limits_{F\in\mathcal{F}} L(F)$.

Пусть B_n множество всех наборов $\tilde{\alpha}=(\alpha_0,\ldots,\alpha_{n-1})$ из нулей и единиц. Отображение B_n в B_m будем называть (n,m) - оператором. Через $|\tilde{\alpha}|$ обозначим $\sum\limits_{i=0}^{n-1}\alpha_i 2^i$ и через $\|\tilde{\alpha}\|$ - число единиц в наборе $\tilde{\alpha}$.

Обозначим через $T_n(n+]\log(n+1)[,n)$ - оператор, который сдвигает направо произвольный набор $\tilde{x}$ на произвольное число $|\tilde{\tilde{y}}|$. /Символ]a[означает наименьшее целое число, не меньшее a/. Более точно оператор T_n по двум наборам $\tilde{x}=(x_0,\ldots,x_{n-1})$ и $\tilde{y}=(y_0,\ldots,y_{]\log(n+1)[-1})$ выдает следующий набор $\tilde{z}=(0,\ldots,0,x_0,\ldots,x_{n-|\tilde{y}|-1})$. Обозначим его через $\tilde{x}_{\rightarrow|\tilde{y}|}$. Известна следующая оценка О.Б. Лупанова сложности реализации оператора T_n:

Л е м м а 1.[2]

$$L(T_n)\le C_T\cdot n\cdot\log n$$

Всюду в этой статье буквой C /с индексами, штрихами и т.д./

обозначаются некоторые константы, символ log означает логарифм по основанию 2 и через ℓ_n будем обозначать величину $]\log(n+1)[$.

О.Б. Лупановым была высказана гипотеза, что оператор T_n не допускает линейной реализации.

Интересной является следующая задача: какова сложность реализации класса $\mathcal{T} = \{T^{\tilde{\alpha}} \mid \tilde{\alpha} = (\alpha_0,\ldots,\alpha_{n-1}) \in B_n\}(\ell_n, n)$ - операторов, которые получаются из оператора T_n подстановками констант $\alpha_0,\ldots,\alpha_{n-1}$ на место переменных $x_0,\ldots,x_{n-1}$?

Здесь доказывается линейность реализации одного класса операторов $T^{\tilde{\alpha}}$, а именно тех, которые определены множеством наборов $\tilde{\alpha}$, у которых, по порядку, не больше $\frac{\log n}{\log\log n}$ единиц.

Введем обозначения: $V_{n,k} = \{\tilde{\alpha} \mid \tilde{\alpha} \in B_n, \|\tilde{\alpha}\| = k\}$,
$\mathcal{T}_A = \{T^{\tilde{\alpha}} \mid \tilde{\alpha} \in A \subseteq B_n\}$

<u>Т е о р е м а</u>. Для любого $k \le C \frac{\log n}{\log\log n}$, $L(\mathcal{T}_{V_{n,k}}) \le C_{T_k} \cdot n$.

Сначала введем вспомогательные операторы, а потом сформулируем и докажем несколько лемм, из которых и будет следовать утверждение теоремы.

1°. Оператор K_m /"дешифратор"/. Это $(m, 2^m)$ - оператор, который преобразует любой набор $\tilde{x} = (x_0,\ldots,x_{m-1})$ в набор $\tilde{y} = (y_0,\ldots,y_{2^m-1})$ такой, что

$$y_i = \begin{cases} 1, & \text{если } i = |\tilde{x}| \\ 0, & \text{если } i \ne |\tilde{x}| \end{cases}$$

Хорошо известен следующий факт.

Л е м м а 2.1. $L(K_m) \le C_K \cdot 2^m$.

2°. Оператор E_m /в некотором смысле является обратным к K_m/. Это $(2^m, m+1)$ - оператор. Он набор $\tilde{\xi}_i = (0,\ldots,0,\underset{i}{1},0,\ldots,0)$

длины 2^m преобразует в двоичную запись номера разряда, в котором стоит единица, нулевой набор преобразует в набор (0,...,0,1), а на остальных наборах он может быть произвольным.

Л е м м а 2.2.[2] $L(E_m) \le C_E \cdot 2^m$.

3^o. Оператор умножения $U_{m,\ell}$. Это $(m+\ell,\ m+\ell)$ - оператор. Он по наборам $\tilde{x}$ длины m и $\tilde{y}$ длины ℓ находи $m+\ell$ разрядов их произведения

$$U_{m,\ell}(\tilde{x},\tilde{y}) = \tilde{z}\ ,\ \text{где}\ \ |\tilde{z}| = |\tilde{x}|\cdot|\tilde{y}|\ .$$

Л е м м а 2.3. $L(U_{m,\ell}) \le C_U \cdot m \cdot \ell$.

Доказательство легко вытекает из обычного "школьного" алгоритма.

4^o. Операторы деления D_m^k определены натуральным числом k, $1 \le k < 2^m$. Для данного k это $(m,\ 2m)$ - оператор, который по набору $\tilde{x}$ длины m выдает наборы: $\tilde{y}$ длины m такой, что $|\tilde{y}| = \left[\frac{|\tilde{x}|}{k}\right]$ и $\tilde{z}$ длины m такой, что $|\tilde{z}| = |\tilde{x}| - |\tilde{y}|\cdot k$. /Символ [a] означает наибольшее целое число - меньше a./

Л е м м а 2.4. Для любого k, $1 \le k < 2^m$,

$$L(D_m^k) \le C_D \cdot 2^m.$$

Доказательство тривиальное.

5^o. Операторы ограниченного сдвига $M^{\tilde{\alpha}}$, $\tilde{\alpha} \in B_n$. Для данного набора $\tilde{\alpha}$, $M^{\tilde{\alpha}}$ это $(\ell_n,\ n)$ - оператор, который произвольный набор $\tilde{x}$ длины ℓ_n преобразует в набор $\vec{\tilde{\alpha}}_{|\tilde{x}|}$, если $|\tilde{x}| < C_{25}\,\frac{n}{\|\tilde{\alpha}\|}$, и в нулевой набор длины n , в остальных случаях.

Л е м м а 2.5. Для любого $\tilde{\alpha} \in B_n$,

$$L(M^{\tilde{\alpha}}) \le C_M \cdot n.$$

Доказательство. Обозначим через p_i, $i = 1,2,\dots,\|\tilde{\alpha}\|$ номера разрядов, в которых находятся единицы в наборе $\tilde{\alpha}$. Схема для $M^{\tilde{\alpha}}$ строится в соответствии со следующим алгоритмом /рис. 1/:

1. Входной набор $\tilde{x}$ подается на входы "дешифратора" K_{ℓ_n}. $L(K_{\ell_n}) \le C'_K \cdot n$ /лемма 2.1/.

2. Младших $C_{25} \frac{n}{\|\tilde{\alpha}\|}$ разрядов выходного набора "дешифратора" /или он целиком, если $\|\tilde{\alpha}\| \le C_{25}$/ подаются на "вторые" входы $\|\tilde{\alpha}\|$ - штук схем A_{p_i}, $i = 1,\dots,\|\tilde{\alpha}\|$, где схема A_{p_i} сдвигает i-ю единицу набора $\tilde{\alpha}$. На "первые" входы схемы A_{p_1} подается нулевой набор длины n, а на "первые" входы схемы A_{p_i}, $i > 1$, подается выходной набор схемы $A_{p_{i-1}}$ /т.е. набор, у которого первых $(i-1)$ единиц уже поставлены на свои места/. Более формально схема A_p, $0 \le p \le n-1$ реализует $(n + C_{25} \frac{n}{\|\tilde{\alpha}\|}, n)$ - оператор, который по наборам $\tilde{\alpha} = (\alpha_0,\dots,\alpha_{n-1})$ и $\tilde{\beta} = (\beta_0,\dots,\beta_{C_{25}\frac{n}{\|\tilde{\alpha}\|}-1})$ вычисляет набор $\tilde{\gamma} = (\gamma_0,\dots,\gamma_{n-1})$ такой, что

$$\gamma_i = \begin{cases} \alpha_i \vee \beta_{i-p}, & p \le i < \min\{n, C_{25} \frac{n}{\|\tilde{\alpha}\|} + p\} \\ \alpha_i, & \text{в остальных случаях.} \end{cases}$$

Очевидно, для любого i, $L(A_{p_i}) \le C_{25} \frac{n}{\|\tilde{\alpha}\|}$.

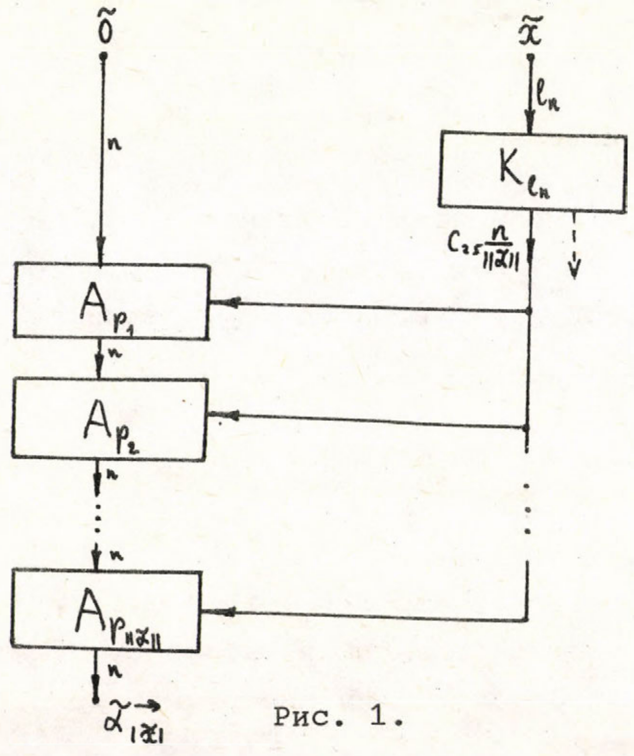

Рис. 1.

Таким образом, $L(M^{\tilde{\alpha}}) \le L(K_{\ell_n}) + \sum_{i=1}^{\|\tilde{\alpha}\|} L(A_{p_i}) \le C_M \cdot n$.

Лемма доказана.

Теперь сформулируем и докажем четыре леммы, из которых и будет следовать утверждение теоремы.

Л е м м а 3. Пусть набор $\tilde{\alpha} \in B_n$ такой, что между любыми двумя соседними единицами в наборе существует хотя бы $(K-1)$ нулевей, где $K \ge C_3' \log n \cdot \log\log n$. Тогда,

$$L(T^{\tilde{\alpha}}) \le C_3 \cdot n .$$

Доказательство. Введем обозначения: $K_1 = [\log K]$, $K' = 2^{K_1}$,
$\tilde{\gamma} = (\gamma_0, \ldots, \gamma_{\ell_n - 1})$ - двоичная запись числа $(K_1 + 1)$ и
$\tilde{y} = (y_0, \ldots, y_{\ell_n - 1})$ - набор, которым задается величина сдвига.
Схема для оператора $T^{\tilde{\alpha}}$ строится в соответствии со следующим алгоритмов /см. рис. 3/:

1. По набору $\tilde{y}$ определяются наборы $\tilde{\sigma}_1$ и $\tilde{\sigma}_2$ - двоичные записи чисел $[\frac{|\tilde{y}|}{K'}]$ и $|\tilde{y}| - [\frac{|\tilde{y}|}{K'}] \cdot K'$. Это делает схема $D^{K'}_{\ell_n}$. $L(D^{K'}_{\ell_n}) \le C_D' \cdot n$ /лемма 2.4./.

2. Набор $\tilde{\sigma}_2$ подается на входы схемы $M^{\tilde{\alpha}}$, которая сдвигает набор $\tilde{\alpha}$ на число $|\tilde{\sigma}_2| < K' \le K \le C'' \frac{n}{\|\tilde{\alpha}\|} \cdot L(M^{\tilde{\alpha}}) \le C_M \cdot n$ /лемма 2.5./.

3. Наборы $\tilde{\sigma}_2$ и $\tilde{\gamma}$ подаются на входы схемы U_{ℓ_n, ℓ_n}, которая их умножает. $L(U_{\ell_n, \ell_n}) \le C_U' \cdot \log^2 n$ /лемма 2.3./.

4. Выходы схемы $M^{\tilde{\alpha}}$ разобъем слева направо на $]n/K'[$ групп, по K' выходов в каждой /кроме, может быть, последней/ и подаем их на входы $]n/K'[$ - штук "кодеров" E_{K_1} /последними, на оставшиеся свободными входы, подаем нули!/. $L(E_{K_1}) \le C_E \cdot K'$ /лемма 2.2/.

5. Объединяя выходы этих "кодеров", их $]n/K'[\cdot (K_1+1)$ штук,

подаем их вместе с $\ell_{]n/K'[\cdot(K_1+1)}$ левыми выходами схемы U_{ℓ_n,ℓ_n} на входы схемы, которая реализует оператор $T_{]n/K'[\cdot(K_1+1)}$, т.е. сдвигает закодированный набор, полученный на выходах "кодеров" E_{K_1}, на число $[\frac{|y|}{K'}](K_1+1)$. $L(T_{]n/K'[\cdot(K_1+1)})\le C'_T\cdot]n/K'[\cdot K_1\cdot \log n$ /лемма 1./.

6. Выходы схемы $T_{]n/K'[\cdot(K_1+1)}$ разобъем слева направо на $]n/K'[$ групп, по (K_1+1) штук в каждой. В каждой группе, левых K_1 выходов подаются на входы дешифратора K_{K_1} и полученный набор на его выходах поразрядно умножается на последний выход в группе. Обозначим череб $\tilde{\beta}$ набор длины n, который получается на левых n из $]n/K'[\cdot K'$, таким образом, полученных выходах. $L(K_{K_1})\le C_K\cdot K'$ /лемма 2.1./.

7. Набор $\tilde{y}$ подается на входы дешифратора $K_{\ell_n}\cdot L(K_{\ell_n})\le C'_K\cdot 2^n$. /лемма 2.1./.

8. Выходы этого дешифратора подаются на входы схемы H_n, которая любой набор вида /0,...,0,1,0,...,0/ длины n преобразует в набор /0,...,0,1,1,...,1/ длины n, а на остальных наборах она произвольная /см. рис. 2/. Очевидно, $L(H_n)\le n$.

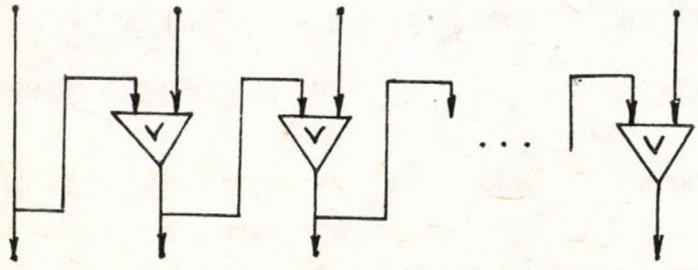

Рис. 2.

9. Набор, реализованный на выходах схемы H_n, поразрядно умножается на набор $\tilde{\beta}$, полученный в шаге 6.

Таким образом, учитывая, что $K \geq C_3' \cdot \log n \cdot \log \log n$, $L(T^{\tilde{\alpha}}) \leq C_3 \cdot n$.

Лемма доказана.

Введем одно понятие. Любой поднабор $(\alpha_i, \alpha_{i+1}, \ldots, \alpha_{i+s})$ набора $\tilde{\alpha} = (\alpha_0, \ldots, \alpha_{n-1})$ называется пустым куском, если все α_j равны 0, $j = i, i+1, \ldots, i+s$. Поднабор, который этому не удовлетворяет, будем называть непустым куском.

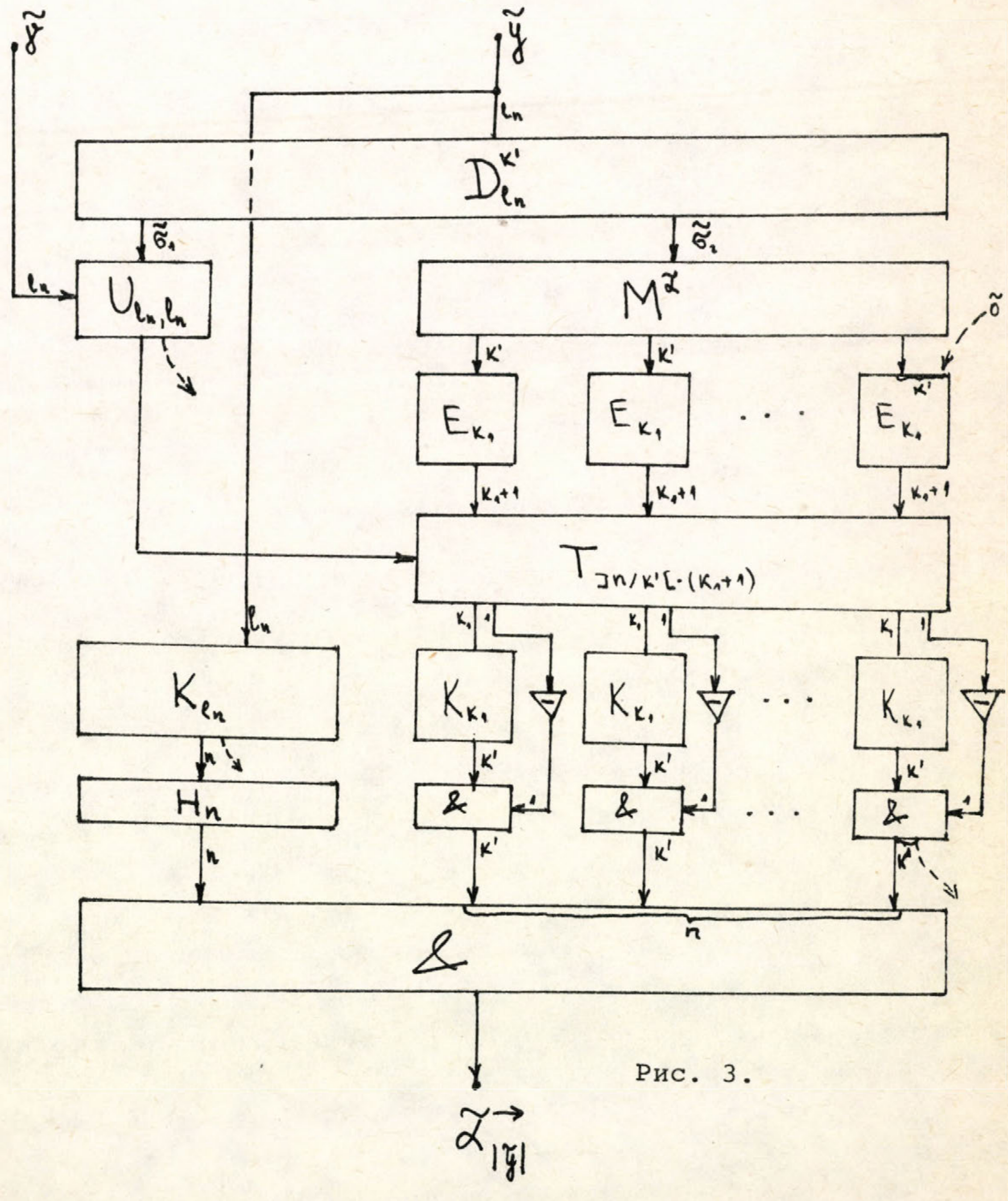

Рис. 3.

Л е м м а 4. Пусть набор $\tilde{\alpha} \in B_n$ такой, что если его рабобъем на куски длины К, $K \le \frac{n}{\log n}$, то его единицы окажутся не более C_4' кусков. Тогда

$$L(T^{\tilde{\alpha}}) \le C_4 \cdot n.$$

Доказательство. Нетрудно убедиться, что можно предположить, что только один кусок длины К в наборе $\tilde{\alpha}$ непустой. Из этого куска и куска такой же длины, но состоящего только из нулей, образуем набор $\tilde{\sigma}$ длины 2К /см. рис. 4./.

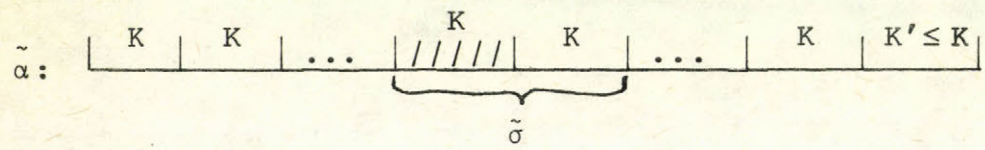

Рис. 4.

Обозначим через $\tilde{y}$ набор, которым задается величина сдвига, через $\tilde{\nu}$ набор длины $]\frac{n}{K}[$, где $\nu_i = 1$ тогда и только тогда, когда i-й кусок набора $\tilde{\alpha}$ непустой. Схема для оператора $T^{\tilde{\alpha}}$ строится в соответствии со следующим алгоритмом /рис. 5./:

1. По набору $\tilde{y}$ определяются наборы $\tilde{\sigma}_1$ и $\tilde{\sigma}_2$ - двоичные записи чисел $[\frac{|\tilde{y}|}{K}]$ и $|\tilde{y}| - [\frac{|\tilde{y}|}{K}] \cdot K$. Это делает схема $D^K_{\ell_n}$. $L(D^K_{\ell_n}) \le C_D' \cdot n$ /лемма 2.4./.

2. Наборы $\tilde{\sigma}$ и младших ℓ_{2K} разрядов набора $\tilde{\sigma}_2$ подаются на входы схемы, которая реализует оператор T_{2K}, т.е. сдвигает набор $\tilde{\sigma}$ на величину $|\tilde{\sigma}_2|$. $L(T_{2K}) \le C_{\mathcal{T}} \cdot 2K \cdot \log(2K)$ /лемма 1./.

3. Младшие $\ell_{]\frac{n}{K}[}$ разряды набора $\tilde{\sigma}_1$ подаются на входы схемы $M^{\tilde{\nu}}$, которая сдвигает набор $\tilde{\nu}$ на величину $|\tilde{\sigma}_1|$. Обозначим этот

сдвинутый набор через $\tilde{\delta} = (\delta_0, \ldots, \delta_{]\frac{n}{K}[-1})$. $L(M^{\tilde{\nu}}) \le C_M \cdot]\frac{n}{K}[$ /лемма 2.5./.

4. С помощью схем операторов B_i, $i = 0, 1, \ldots,]\frac{n}{K}[-1$ "поставим" набор, реализованный на выходе схемы T_{2K} на место, которое соответствует величине сдвига /оно определено положением единицы в наборе $\tilde{\delta}$ /. Формально, B_i это $(n+2K+1, n)$ - оператор, который по наборам $\tilde{\gamma}$ длины n, $\tilde{\sigma}'$ длины 2K и разряду δ_i определяет набор $\tilde{z} = (z_0, \ldots, z_{n-1})$ такой, что

$$z_j = \begin{cases} \gamma_j \vee \sigma'_{j-i\cdot K} \& \delta_i, & i\cdot K \leqslant j < \min\{n, (i+2)\cdot K\} \\ \gamma_j, & \text{в остальных случаях} \end{cases}$$

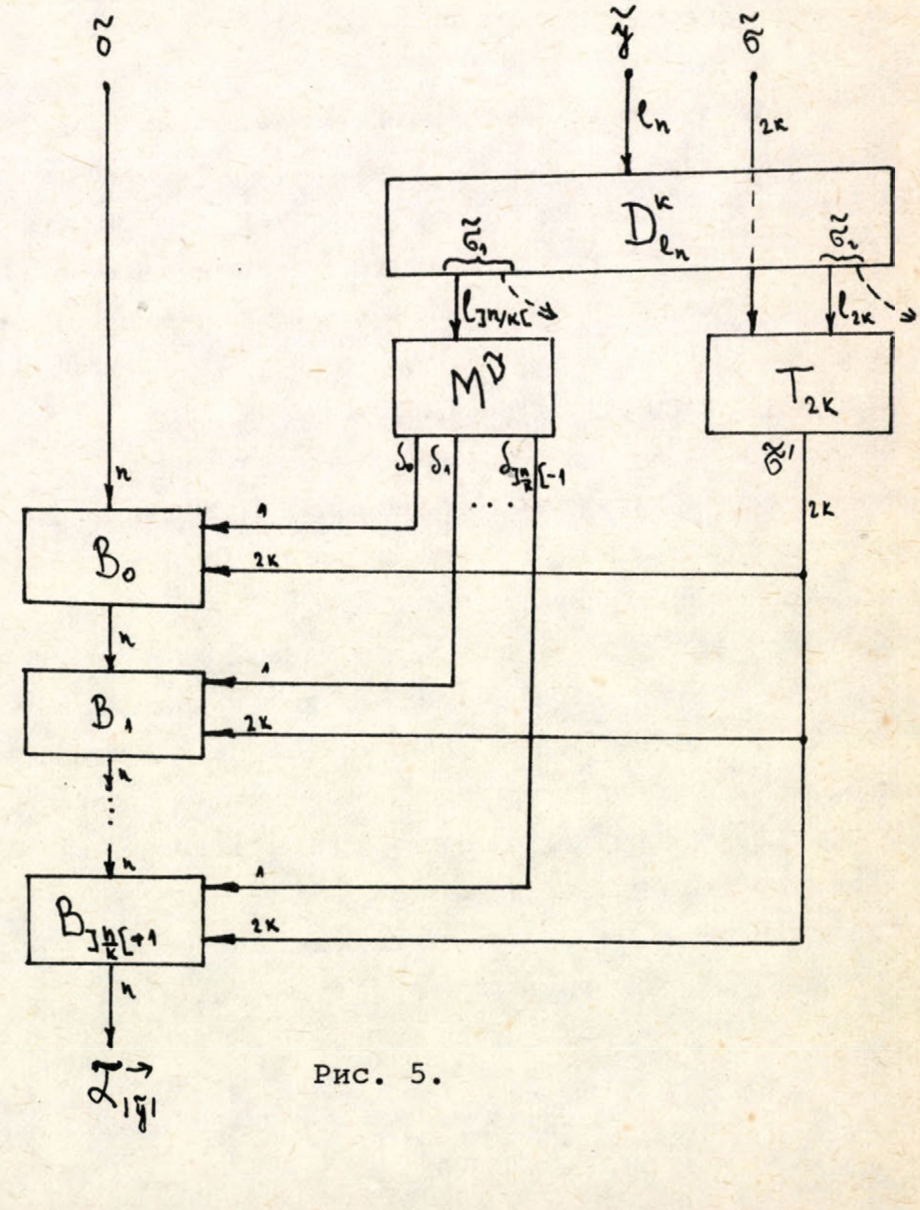

Рис. 5.

Очевидно, что для любого i, $L(B_i) \le C_B \cdot K$. Таким образом,

$$L(T^{\tilde{\alpha}}) \le C'_D \cdot n + C_T \cdot 2K \cdot \log(2K) + C_M]\frac{n}{K}[+]\frac{n}{K}[\cdot C_B \cdot K \le C_4 \cdot n \ .$$

Лемма доказана.

Л е м м а 5. Пусть набор $\alpha \in B_n$ можно разбить на куски длины K /кроме последнего/, $K \ge C'_5 \cdot \log n \cdot \log\log n$, так чтобы любыми двумя соседними непустыми кусками, пустых кусков было больше, чем единиц в правом из этих двух непустых кусков. Тогда

$$L(T^{\tilde{\alpha}}) \le C_5 \cdot n \ .$$

Доказательство. Пусть набор $\tilde{\alpha}$ удовлетворяет данному условию. Будем преобразовывать набор $\tilde{\alpha}$, сдвигая его единицы так, что получим набор $\tilde{\beta}$, который удовлетворяет условию леммы 3.

Рассмотрим произвольный непустой кусок в наборе $\tilde{\alpha}$. Обозначим через ℓ число единиц в нем. По условию леммы, левее от этого куска существует $\ell+1$ пустых кусков. Единицы из непустого куска разместим по пустым кускам следующим спомобом: последнюю /если смотреть справа налево/ сдвинем влево на K разрядов, предпоследнюю – на 2K разрядов и т.д., ℓ-ю на $\ell \cdot K$ разрядов. Если это сделаем с любым непустым куском, получим набор $\tilde{\beta}$. Разобъем его на куски длины K. Он обладает следующими свойствами: в каждом куске нет больше одной единицы; расстояние между соседними единицами >K; между соседними единицами, которые принадлежали одному куску в наборе $\tilde{\alpha}$, нет пустых кусков, а между соседними единицами, которые принадлежали разным кускам в наборе $\tilde{\alpha}$ существуют хотя бы два пустых куска. Схему будем строить так, что сначала будем сдвигать набор $\tilde{\beta}$, а потом инверсно описанному выше алгоритму, восстанавливать сдвинутый набор $\tilde{\alpha}$. Ради однозначности декодировки сдвинутого набора "удлиним" набор $\tilde{\beta}$ направо на нулевой кусок длины $(]\frac{n}{K}[+1) \cdot K-n$. Полученный набор обозначим через $\tilde{\beta}'$. Теперь опишем алгоритм, в соответствии с которым будем строить схему для оператора $T^{\tilde{\alpha}}$ /см. рис. 8./:

1. Набор $\tilde{y}$, которым задается величина сдвига, подается на выходы схемы, которая реализует оператор $T^{\tilde{\beta}'}$. $L(T^{\tilde{\beta}'}) \le C_3 \cdot n$ /лемма 3./.

2. Выходы этой схемы разобъем на $]\frac{n}{K}[+1$ зон, по K - штук в каждой. Обозначим через $\tilde{\nu}_i$ набор, который реализуется в i-ой зоне, $i = 1,2,\ldots,]\frac{n}{K}[+1$. Наборы $\tilde{\nu}_i$, $i = 1,\ldots,]\frac{n}{K}[+1$ подаются на столько же дизъюнктивных схем V_i, которые обнаруживают присутствие единицы в i-ой зоне, т.е. вычисляются сигналы $t_i = \bigvee\limits_{j=0}^{K-1} \nu_{ij}$. Очевидно, для любого i, $L(V_i) \le K$.

3. По сигналам t_j, $]\frac{n}{K}[$ штук схем R_i вычисляют сигналы r_i, где $r_i = 0$ в том случае, когда единицу из i-ой зоны надо отнести в левый собирающий кусок /см. шаг 5. и рис. 5./, и $r_i = 1$, если эту единиуц надо отнести в правый собирающий кусок /в случае, когда непустой кусок набора $\tilde{\alpha}$ при сдвиге частично попал в два куска длины K/. Нетрудно убедиться, что должно быть: $r_1 = 0$ и $r_{i+1} = \bar{r}_i \cdot t_i \cdot \bar{t}_{i+1} \cdot t_{i+2} \vee r_i t_{i+1}$, $i = 1,2,\ldots,]\frac{n}{K}[-1$. Таким образом, $L(R_i) \le C_R$ для любого i.

4. По сигналам t_s и r_ℓ, $]\frac{n}{K}[$ штук схем P_i, $i = 1,\ldots,]\frac{n}{K}[$ вычисляют сигналы p_i, где $p_i = 1$, когда в i-й и $(i+1)$-й кусок надо поставить левый и правый собирающий кусок /потому, что единицы "вернулись" на свое место/ и $p_i = 0$ в остальных случаях. Нетрудно убедиться, что должно быть $p_i = \bar{t}_i \cdot \bar{t}_{i+1} \vee t_i \cdot \bar{t}_{i+1} \cdot r_i$, $i = 1,\ldots,]\frac{n}{K}[$ откуда и следует оценка $L(P_i) \le C_p$ для любого i.

5. Для каждого i, $i = 1,\ldots,]\frac{n}{K}[$ строится схема W_i, которая определяет набор $\tilde{\sigma}_i$ длины $2K$, т.е. левый и правый собирающий кусок. Более подробно, схема W_i зависимо от значения сигнала r_i, пропускает единицу из i-ой зоны /набора $\tilde{\nu}^i$/ в левый или правый кусок набора $\tilde{\sigma}_{i-1}$, умноженного сначала поразрядно на p_{i-1} /см. рис. 6./. Для $\tilde{\sigma}_0$ берется нулевой набор длины $2K$, $p_0 = 1$.

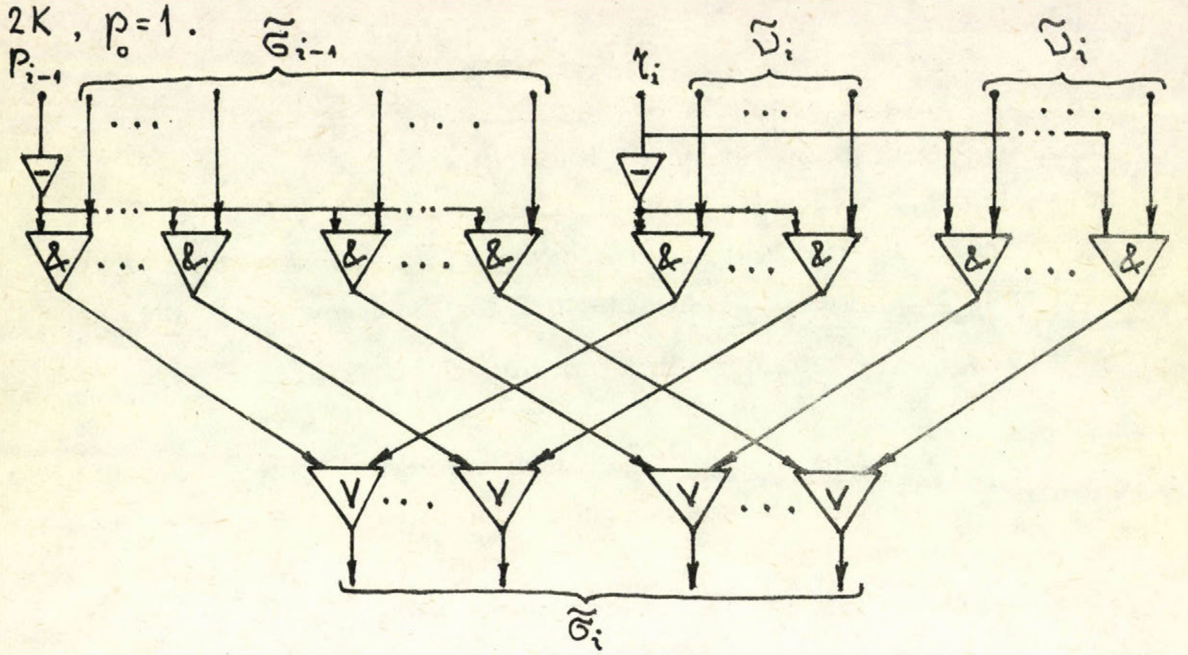

Рис. 6.

Очевидно, что $L(W_i) \le C_W \cdot K$ для любого i.

6. Наборы $\tilde{\sigma}_i$, $i = 1,\ldots,]\frac{n}{K}[$ умножаются поразрядно на p_i при помощи $]\frac{n}{K}[$ штук схем G_i. Очевидно, что $L(G_i) \le C_G \cdot K$ для любого i.

7. Обозначим наборы, которые получаются на выходах схемы G_i через $\tilde{\sigma}'_i$, $i = 1,\ldots,]\frac{n}{K}[$. При помощи $]\frac{n}{K}[-1$ штук схем Σ_i будем поразрядно суммировать правый кусок набора $\tilde{\sigma}'_i$ с левым куском набора $\tilde{\sigma}'_{i+1}$ /см. рис. 7./.

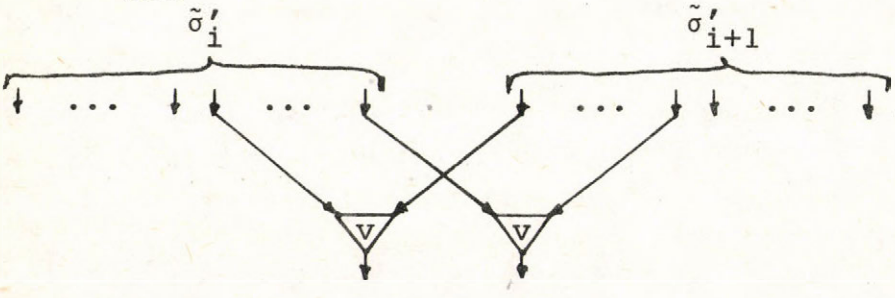

Рис. 7.

Очевидно, что $L(\Sigma_i) \le C_\Sigma \cdot K$ для любого i. Объединяя оставшиеся выходы схемы G_1 с выходами схемы Σ_i, $i = 1,\dots,]\frac{n}{K}[-2$ и $n-(]\frac{n}{K}[-1)\cdot K$ выходов схемы $\Sigma_{]\frac{n}{K}[-1}$, получим на этих выходах набор $\tilde{\alpha}^{\to}_{|\tilde{y}|}$.

Таким образом, $L(T^{\tilde{\alpha}}) \le C_6 \cdot n$.

Лемма доказана.

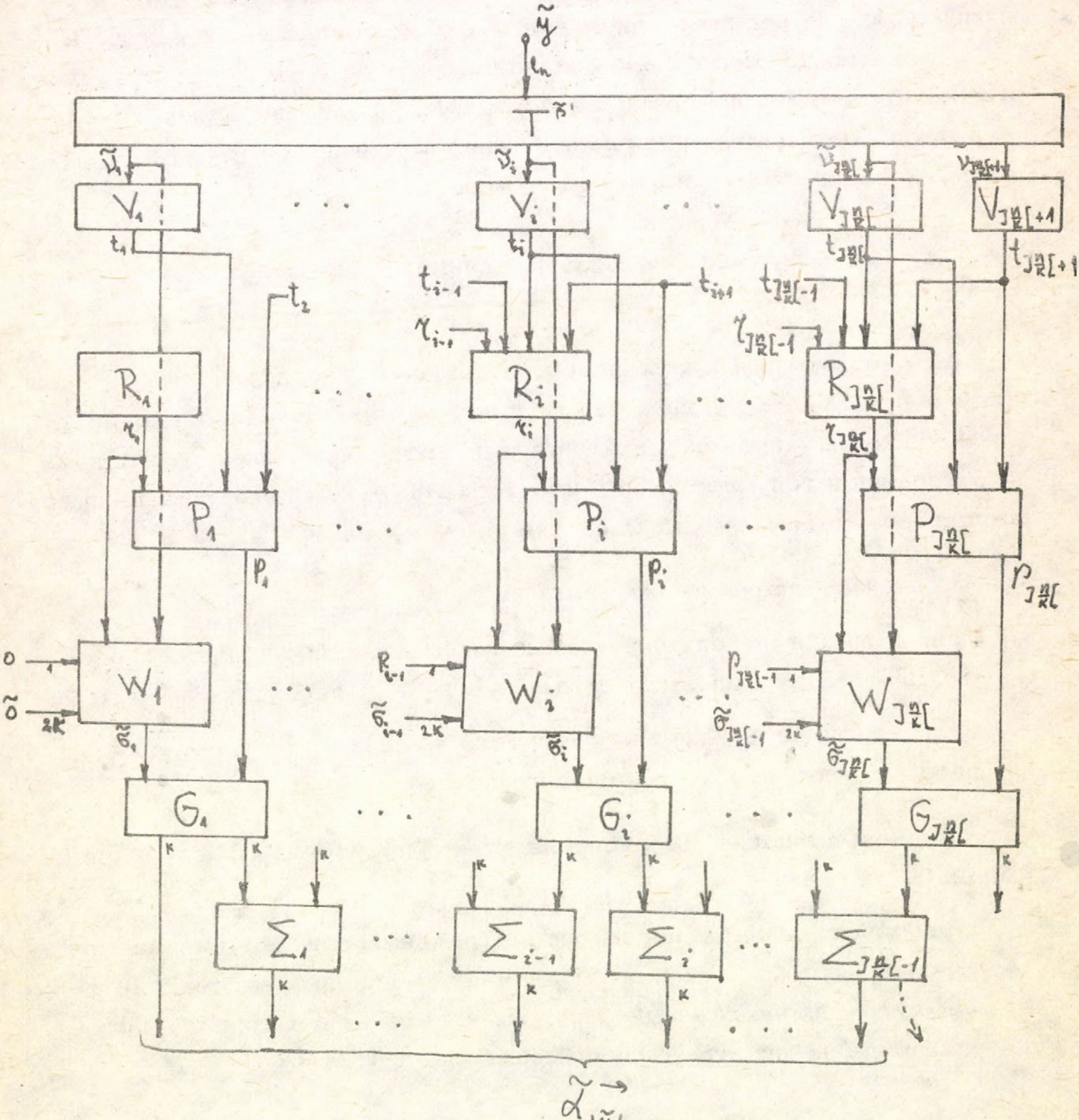

Рис. 8.

Л е м м а 6. Любой набор $\tilde{\alpha} \in B_n$, $\|\tilde{\alpha}\| = K$, можно разложить в поразрядную сумму по mod2 двух наборов: один удовлетворяет условию леммы 5, а у второго число разрядов между первой и последней единицами $< K^K \cdot M$, где $M = \log n \cdot \log\log n$.

Доказательство. Разобъем единицы в наборе $\tilde{\alpha}$ на группы так, чтобы между соседними единицами в одной группе было меньше M разрядов, между соседними единицами разных групп - $\geq M$ разрядов. Очевидно, что число групп не больше K. Обозначим длину /число разрядов/ самой длинной группы через D_1. Теперь приведем алгоритм, которым будем объединять "меньшие" группы в "большие".

1. Рассмотрим самую правую единицу в наборе $\tilde{\alpha}$ и положим $i = 1$.

2. Рассмотрим кусок набора $\tilde{\alpha}$ длины D_i, содержащий эту единицу и D_{i-1} разряд левее ее. Обозначим через p число единиц в этом куске, а через r число "пустых" разрядов левее этого куска. Спрашивается, верно ли, что $r \geq (p+1)D_i$? Если неравенство выполнено, переходим на шаг 3, если нет, на шаг 4.

3. У нас две возможности:

а/ если существуют единицы левее рассматриваемого куска, рассмотрим первую из них /справа налево/ и переходим на шаг 2;

б/ если не существуют, алгоритм останавливается и, очевидно, набор $\tilde{\alpha}$ удовлетворяет условию леммы 5.

4. Невыполнение неравенства может произойти по двум причинам:

а/ существует единица левее рассматриваемого куска, из-за которой $r < (p+1)D_i$. В этом случае, "удлиним" рассматриваемый кусок влево пока не включим в него целую группу, которой принадлежит эта единица /см. рис. 9./.

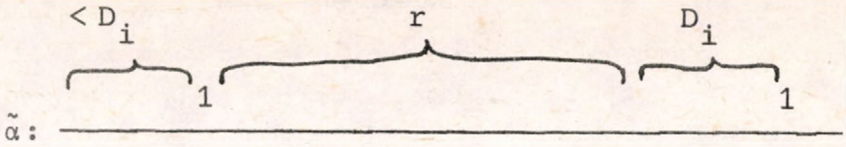

Рис. 9.

Длину полученного таким образом куска обозначим через D_{i+1}. Очевидно, $D_{i+1} < D_i + r + D_i < (p+3)D_i$. Опять возвращаемся к самой правой единице в наборе $\tilde{\alpha}$, но уже с новым D_i /i на единицу больше/ и переходим на шаг 2.

б/ Не существует единиц левее рассматриваемого куска. Алгоритм останавливается.

Таким образом, алгоритм останавливается либо в шаге 3 б/ , и набор $\tilde{\alpha}$ удовлетворяет условию леммы 5., либо в шаге 4 б/ , когда набор $\tilde{\alpha}$ можно поразрядно разложить на сумму по mod2 набора, который удовлетворяет условию леммы 5. /это правый кусок набора или нулевой набор/ и набора, у которого между крайними единицами $< D_j$ разрядов, для некоторого j. Нетрудно заметить, что при каждом выполнении шага 4 а/ алгоритма, число групп в наборе $\tilde{\alpha}$ уменьшается хотя бы на единицу. Следовательно $j < K$. Из $D_{i+1} < (p+3)D_i$ и $D_1 < K \cdot M$ получим $D_j < K^K \cdot M$ для любого j.

Лемма доказана.

Доказательство теоремы непосредственно вытекает из лемм 4., 5. и 6. и факта, что при $K \le \dfrac{\log n}{\log\log n}$ выполнено неравенство $K^K \cdot M < C' \dfrac{n}{\log n}$.

ЛИТЕРАТУРА

[1] Лупанов О.Б.: О синтезе некоторых классов управляющих систем. В сб.: Проблемы кибернетики, М., 1963, вып. 10, стр. 63-97.

[2] Лупанов О.Б.: Об одном подходе к синтезу управляющих систем - принципе локального кодирования. - В сб.: Проблемы кибернетики, М., 1965, вып. 14, стр. 31-110.

Bizonyos eltolás operátorok realizációinak lineáris komplexitása

R.L. Ščepanovič

Összefoglaló

Jelöljük $L(F)$-el az F operátor komplexitását /l. [1]/, és $L(\mathcal{F}) = \max_{F\in\mathcal{F}} L(F)$ jelentse az $\mathcal{F}$ operátorcsalád komplexitását. Lupanov [1] belátta, hogy bizonyos fajta T_n eltolás operátorra $L(T_n) \leq C\, n\cdot\log n$ és azt sejtette, hogy a T_n operátornak nincs olyan realizációja, amelynek komplexitása lineáris volna. A cikkben a szerző bemutat egy olyan $\mathcal{T}$ operátor családot, amely T_n operátoroknak bizonyos módositásait tartalmazza és amelyre $L(\mathcal{T}) \leq c\, n$.

Linear complexity of some translation operators

R.L. Ščepanovič

Summary

Denote by $L(F)$ and $L(\mathcal{F}) = \max_{F\in\mathcal{F}} L(F)$ the complexity of the operator F and the operator family $\mathcal{F}$ respectively. (see [1]). Lupanov [1] proved that $L(T_n) \leq c\, n\cdot\log n$ for some translation operators T_n and conjectured that T_n has no linear realization. In the paper a family $\mathcal{T}$ of modified operators T_n is shown, such that $L(\mathcal{T}) \leq c\, n$.