MULTIPLICATIVE LINEAR LOGIC FOR RESOURCE TRANSFORMATION NETS

ALEXANDER M. DEKHTYAR

Tver State University, Russia
p. VNIISV 41, apt. 30., Tver 32, 170032, Russia

ABSTRACT.
In this paper we introduce a new type of nets with distributed resources : resource transformation nets (RT-nets) . A new calculus based on Horn fragment of Multiplicative Linear Logic is proposed for this class of models. Theorem of completness is proved.

1. INTRODUCTION

The idea of relating the concurrent computations to the linear logic proofs was first proposed by Girard [3]. A number of authors considered Petri nets and some other kinds of nets for representing concurrency in connection with linear logic [1,2,4].

In this paper a new class of nets called RT-nets (resource transformation nets) is proposed. An RT-net consists of an oriented graph and a function that for each node of graph and for each type of resource returns a maximum number of copies of a resource that can be initially stored in that node. A resource can be either from a finite set $S = \{s_i, ..s_k\}$ (resources of this kind are called supplies) or a construction $(X \rightarrow Y)$ from a finite set P, where X and Y are lists of elements of S (these constructions are called basic converters). Suppose we are given an initial set of resources A and a final set of resources B where each single resource of B is marked with a node of the graph. Such pairs (A, B) will be called further requests. The problem that is to be solved on RT-nets is : is it possible to distribute resources A on the graph so that after an execution process each resource from B would be contained in the node that marks this resource ? An execution of the net is divided on steps. Steps of execution can be of two different kinds. First kind of step is resource transformation : if a node contains X and $(X \rightarrow Y)$ then we can replace this pair with Y. Second kind of step is resource transmission: if node α contains resource Y and (α, β) is an edge of the graph then we can transmit Y from α to β. An initial distribution of A on nodes of the graph (it is called initialization) should be performed with the respect to a restriction function.

Informaly RT-nets can be thought of in the following way. Suppose the graph represents a factory where nodes are representations of its divisions and edges represent the connections between divisions. Supplies can be thought of as materials on which the factory operates and converters represent technological processes :

$(X \rightarrow Y)$ means that a set of resources X can be processed into production (materials) Y. If a technological process is associated with a division it means the presence of equipment for this process in this division. A restriction function determines the storage capacities of each division for every type of material and a number of times each technological process can be run in every division during one industrial cycle. Now, a pair (A, B) represents an industrial process, where A is a set of initial materials for the process and B is a set of resulting products where for which single product a division in which it is to be finally stored is given. Now we investigate which industrial processes can be run on this factory, i.e. for which pairs (A, B) we can find a distribution of resources from A such that the structure of the factory would allow to process them in one industrial cycle and produce B. Finally, adding elements of concurrency to this scheme we can pose a question : which groups of processes $\{(A_i, B_i)\}$ can be run simulteniously (concurrently) on a given factory ? Or, transforming it into RT-nets terminology, which groups (sets) of requests are successful on a given RT-net ?

In order to answer this question we construct for each RT-net $\mathfrak{M}$ a calculus $T_{\mathfrak{M}}$. Elements of such calculi are sequents $A \vdash B$ where (A, B) a request. In this paper we prove soundness and completeness theorems for $T_{\mathfrak{M}}$ with the respect to RT-net semantics.

RT-nets is a generalization of another type of nets, called BR-nets (bounded resource nets) introduced in [1]. The main difference between these two kinds of nets is that in BR-nets it is not allowed to separate the resources obtained as a result of a transformation, i.e. if Y is a result of transformation $X(X \rightarrow Y)$ in some node of a BR-net, then on all further steps all resources of Y should go together. Such operation, however is possible in RT-net semantics. Syntax of proposed calculus $T_{\mathfrak{M}}$ also extends the syntax of calculus $HT_{\mathfrak{M}}$ from [1] where all sequents of a form $A \vdash B^{(\alpha)}$ with the only one goal node are considered. In **Section 3** we propose a simple example of a net $\mathfrak{M}$ and a sequent which is provable in $T_{\mathfrak{M}}$ and not provable in $HT_{\mathfrak{M}}$.

Section 2 of the paper contains the formal definitions of RT-nets and their functioning. **Section 3** contains the definition of $T_{\mathfrak{M}}$ calculus for an RT-net M and an example of an RT-netand a provable in $T_{\mathfrak{M}}$ sequent. In **Section 4** we formulate some properties of provable sequents and sequences of sequents and the theorems of soundness and completness for $T_{\mathfrak{M}}$ with the respect to an RT-net $\mathfrak{M}$ are proved.

2. RT-Nets

We fix a finite set S. The elements of S will be called supplies. The set *Slist* of supply lists is the least satisfying the following conditions: $S \subseteq \mathit{Slist}$; if $A, B \in \mathit{Slist}$, then $(AB) \in \mathit{Slist}$. Two lists of supplies $A, B \in \mathit{Slist}$ are equivalent iff the multisets of elements of S included in A and B are equal. For example, for $S = \{a, b, c\}$, $((a((bc)a)$ is equivalent to $(a((ba)c))$ but is not equivalent to $(a(bc))$. We do not distingush a list $A \in \mathit{Slist}$ from a multiset of elements of S included in A. An exponential notation will be used for representing multisets. For example a^2b^3c will correspond to a multiset $\{a, a, b, b, b, c\}$ and to any list constructed of these and only these letters. Let $[A]r$ denote the number of copies of element r in multiset (list) A. For example $[((ab)(bc))]b = [ab^2c]b = 2$.

Let X and Y be supply lists. Then an expression $(X \rightarrow Y)$ is called a converter. We fix a finite set P of converters. The elements of P will be called basic converters. Let $R = S \cup P$. R is said to be a set of basic resources.

RT-nets (resource transformation nets) are intended to describe resource transmissions and transformations in systems with distributed resources.

Definition 2.1.. *An RT-net $\mathfrak{M}$ is a pair $< G, f >$, where $G = (V, E)$ is a directed graph with a set of nodes V and a set of edges E and $f : V \times R \rightarrow N$ - is a function called restriction.*

For each $\alpha \in V$ and $r \in R$ $f(\alpha, r)$ is a maximum number of copies of resource r that can be stored initialy at the node α.

Definition 2.2.. *A request Q is a pair (A, B) where $A \in Rlist$ and*

$$B = B_1^{\alpha_1} \dots B_k^{\alpha_k}$$

where $(\forall i \in 1 \dots k)(B_i \in Rlist \ \& \ \alpha_i \in V)$.

Further we will unite all resources with the same mark and put them into one subformula, which will give us the right to suppose that $\alpha i \neq \alpha_j$ for $i \neq j$.

Informally, the first element of a request can be thought of as an initial set of supplies and converters. The second element of a request is a goal that is to be obtained from initial set of resources. The nodes of graph which mark parts of this goal show the places of the net where these parts are to be obtained.

An RT-net functions on a set of requests. Its functioning consists of two stages : initialization and execution. On the stage of initialization, RT-net resources are being distributed among the concurrent requests from the set of requests. Execution consists of steps on which the distributed supplies get transformed by applying converters to them or get transmitted between the connected nodes.

Let further Ω denote a set of requests $(Q_1 \dots Q_n)$.

D. *EFINITION 2.3 An initialization of an RT-net calM for a set of requests Ω is a function $f_{init} : V \times R \times Q \rightarrow N$ such that the following conditions hold :*

(i) sufficiency of resources in each node :

$$(\forall \alpha \in V)(\forall r \in R)(\sum_{i=1}^{n} f_{init}(\alpha, r, Q_i) \leq f(\alpha, r))$$

(ii) completness of resource reservation for each request :

$$(\forall Q_i \in Q)(\forall r \in R)(\sum_{\alpha \in V} f_{init}(\alpha, r, Q_i) = [A]r$$

Given any node $\alpha \in V$ and request $Q_i \in \Omega$ let $R(\alpha, i)$ at every moment denote a set of resources associated with Q_i in the node α.

Let $R_{init}(\alpha, i) = \{r^{f_{init}(\alpha, i, Q_i)}\}$ for every request Q_i from Ω. Initially, we put $R(\alpha, i) = R_{init}(\alpha, i)$. The further transformations on $R(\alpha, i)$ are given in **Definition 2.4.**

Definition 2.4.. *An execution of $\mathfrak{M}$ on a set of requests $\mathfrak{Q}$ with an initialization f_{init} is a sequence $\sigma = s_1, \ldots, s_l, \ldots$, of steps of execution.*

A step of execution can be of two different kinds :

(a) Resource transformation.

Let for some $i \in 1 \ldots n$ and for some $\alpha \in V R(\alpha, i)$ contain $\{(X \rightarrow Y)X\}$. The step results in replacing $X(X \rightarrow Y)$ in $R(\alpha, i)$ with Y, i.e. new $R(\alpha, i) := (R(\alpha, i) \backslash \{X(X \rightarrow Y)\}) \cup \{Y\}$ (i.e. this step consists of applying the converter $(X \rightarrow Y)$ to the resource X in the node α) The step does not affect any other $R(\alpha, i)$ sets.

(b) Resource transmission.

Let for some $i \in 1 \ldots n$ and for some $\alpha \in V$ $R(\alpha, i)$ contain $\{X\}$ and let $(\alpha, \beta) \in E$, for some $\beta \in V$. The step results in excluding X from $R(\alpha, i)$ and adding it to $R(\beta, i)$ (i.e. in transmitting the resource X from node α to node β). The step changes the sets :

$$R(\alpha, i) := R(\alpha, i) \backslash \{X\}$$

$$R(\beta, i) := R(\beta, i) \cup \{X\}$$

The step does not affect any other $R(\alpha, i)$ sets.

Definition 2.5.. *A state C of an RT-net on a set of requests $\mathfrak{Q}$ with an initialization f_{init} is a set : $C = \{R(\alpha, i) | \alpha \in V, i \in 1 \ldots n\}$.*

We write $C_1 s C_2$, where C_1 and C_2 are states and s is a step of execution iff step s transforms state C_1 into C_2. We also define and initial state C_{init} for an initialization f_{init} :

$$C_{init} = \{R_{init}(\alpha, i)\}$$

Given an execution $\sigma = s_1, \ldots, s_l$ we write $C_1 \sigma C_2$ iff starting in state C_1 after completing an execution σ net will end up in state C_2.

Definition 2.6.. *A set of requests $\mathfrak{Q} = (Q_1, \ldots, Q_n)$, $Q_i = (A_i, B_i)$, $B_i = B_{i1}^{\beta i1} \ldots B_{ik}^{\beta ik}$ is successfull for an RT-net $\mathfrak{M} =< G, f >$ iff there exists such an initialization f_{init} of $\mathfrak{Q}$ on $\mathfrak{M}$ and there exists such an execution σ of $\mathfrak{M}$ on $\mathfrak{Q}$ that $C_{init} \sigma C_{fin}$ where $C = \{R(\alpha, i) | \alpha \in V, i \in 1 \ldots n\}$, and*

$$R(\alpha, i) = \begin{cases} B_{ij}, & \textit{iff } \alpha = \beta_{ij} \\ \emptyset, & \textit{otherwise} \end{cases}$$

3.Syntax

For every RT-net $\mathfrak{M} =< G, f >, G = (V, E)$, we define a calculus $T_{\mathfrak{M}}$.

Definition 3.1.

1. Alphabet of $T_{\mathfrak{M}}$ is $S \cup \{\vdash, (,), \rightarrow\}$

2. The set of formulas of $T_{\mathfrak{M}}$ is Rlist.

3. For any $r \in R$ and for any $\alpha \in V$ $r^{(\alpha)}$ is a marked formula. If A and B are marked formulas then (AB) is a marked formula. Marked formula $x^{(\alpha)} y^{(\alpha)}$ can be written as $(xy)^{(alpha)}$. Note also that $x^{(\alpha)}x^{(\beta)}$, $(x^2)^{(\alpha)}$ and $(x^2)^{(\beta)}$ are different marked formulas.

4. A sequent is a construction $A \vdash B$ *where* A *is a formula and* B *is a marked formula.*

5. A set of axioms of $T_{\mathfrak{M}}$ *contains* $f(\alpha, r)$ *copies of a sequent* $r \vdash r^{(\alpha)}$*. for each resource* $r \in R$*.*

6. Inference rules :

$$(1)(impl)\quad \frac{A \vdash B^{(\alpha)}D \quad C \vdash (B \to F)^{(\alpha)}H}{AC \vdash F^{(\alpha)}DH}$$

$$(2)(conj)\quad \frac{A \vdash B \quad C \vdash D}{(AC) \vdash (BD)}$$

$$(3)(equ)\quad \frac{A \vdash B}{C \vdash D}$$

if A *is equivalent to* C *and* B *is equivalent to* D*;*

$$(4)(trans)\quad \frac{A \vdash B^{(\alpha)}D}{A \vdash B^{(\beta)}D}$$

if $(\alpha, \beta) \in E$*, and* $B \in Slist$*.*

We will consider the proofs of sequences of sequents from T .

Definition 3.2.. *A proof in* $T_{\mathfrak{M}}$ *is a sequence of sequences of sequents. All the sequents from the first sequence of sequents are distinct axioms of* $T_{\mathfrak{M}}$*. The next sequence of sequents is obtained from previous one by applying one of the proof rules to one or two sequents.*

Definition 3.3.. *A sequence of sequents is called provable in* $T_{\mathfrak{M}}$ *if there exists such a proof in* $T_{\mathfrak{M}}$ *that has the given sequence as its last sequence of sequents.*

Example

Consider the following example of an RT-net $M =< G, f >$. Let $S = \{a, b, c\}$, $P = \{(a \to b^2), (ab \to c)\}$. Let the graph G be as it is shown on the Figure 1. Let the function f be determined by the following table:

$f:$

	a	b	c	$(a \to b^2)$	$(ab \to c)$
α	2	0	0	2	1
β	1	0	0	0	1
γ	0	1	0	0	1
δ	1	0	0	0	1

$G:$

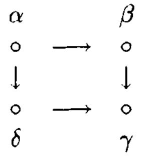

Figure 1.

f determines the set of axioms of T and restrictions on an initial distribution of resources.

Let us consider a request $Q = (a^3(ab \to c)^2(a \to b^2), (c^2)^{(\gamma)}$ and put $Q = (Q)$. We fix $R_{init}(\alpha, 1) = \{a, (a \to b^2)\}$, $R_{init}(\beta, 1) = R_{init}(\delta, 1) = \{a, (ab \to c)\}$, $R_{init}(\gamma, 1) = \emptyset$. Then the execution which begins with the transformation in α

and then transmits one b to β and one b to δ and uses them to obtain c in each of these two nodes and then transmits each c to γ is successful for $\mathbf{\Omega}$ on $\mathfrak{M}$. It is not possible to construct such an execution in BR-net semantics [1].

Now consider the sequent

$$a^3(ab \to c)^2(a \to b^2) \vdash (c^2)^{(\gamma)}.$$

The following proof in $T_{\mathfrak{M}}$ shows that the given sequent is provable :

$$a \vdash a^{(\alpha)} \;\; (a \to b^2) \vdash (a \to b^2)^{(\alpha)} \;\; a \vdash a^{(\beta)} \;\; (ab \to c) \vdash (ab \to c)^{(\beta)} \;\; a \vdash a^{(\delta)} \;\; (ab \to c) \vdash (ab \to c)^{(\delta)}{}_{(1)}$$

$$a(a \to b^2) \vdash (bb)^{(\alpha)} \;\; a \vdash a^{(\beta)} \;\; (ab \to c) \vdash (ab \to c)^{(\beta)} \;\; a \vdash a^{(delta)} \;\; (ab \to c) \vdash (ab \to c)^{(\delta)}{}_{(4)}$$

$$a(a \to b^2) \vdash b^{(\alpha)} b^{(\delta)} \;\; a \vdash a^{(\beta)} \;\; (ab \to c) \vdash (ab \to c)^{(\beta)} \;\; a \vdash a^{(delta)} \;\; (ab \to c) \vdash (ab \to c)^{(\delta)}{}_{(4)}$$

$$a(a \to b^2) \vdash b^{(\beta)} b^{(\delta)} \;\; a \vdash a^{(\beta)} \;\; (ab \to c) \vdash (ab \to c)^{(\beta)} \;\; a \vdash a^{(delta)} (ab \to c) \vdash (ab \to c)^{(\delta)}{}_{(2)}$$

$$a^2(a \to b^2) \vdash (ab)^{(\beta)} b^{(\delta)} \;\; (ab \to c) \vdash (ab \to c)^{(\beta)} \;\; a \vdash a^{(delta)} \;\; (ab \to c) \vdash (ab \to c)^{(\delta)}{}_{(1)}$$

$$a^2(a \to b^2)(ab \to c) \vdash c^{(\beta)} b^{(\delta)} \;\; a \vdash a^{(delta)} \;\; (ab \to c) \vdash (ab \to c)^{(\delta)}{}_{(2)}$$

$$a^3(a \to b^2)(ab \to c) \vdash c^{(\beta)} (ab)^{(\delta)} \;\; (ab \to c) \vdash (ab \to c)^{(\delta)}{}_{(1)}$$

$$a^3(a \to b^2)(ab \to c)^2 \vdash c^{(\beta)} c^{(\delta)} \quad (4)$$

$$a^3(a \to b^2)(ab \to c)^2 \vdash c^{(\beta)} c^{(\gamma)}{}_{(4)}$$

$$a^3(a \to b^2)(ab \to c)^2 \vdash (cc)^{(\gamma)}$$

4. Provability

In this section we establish some properties of proofs in $T_{\mathfrak{M}}$ and prove soundness and completeness theorems.

Lemma 4.1.. *(i) If a sequent $A \vdash B$ is provable then A is not empty (ii) If a sequent $(B \to C) \vdash D$ is provable then D is equivalent to $(B \to C)^{(\alpha)}$ for some $\alpha \in V$. (iii) If a sequence of sequents is provable in $T_{\mathfrak{M}}$ then any its subsequence is provable.*

Lemma 4.2.. *If a sequent has a proof in $T_{\mathfrak{M}}$ then it has a proof in $T_{\mathfrak{M}}$ where in every appearance of rule (1)*

$$\frac{A \vdash B^{(\alpha)} D \quad C \vdash (B \to F)^{(\alpha)} H}{AC \vdash F^{(\alpha)} DH}$$

the sequent $C \vdash (B \to F)^{(\alpha)} H$ is equivalent to the axiom $(B \to F) \vdash (B \to F)^{(\alpha)}$.

Lemma 4.3.. *If a sequent of the form $A \vdash B(C \to D)^{(\alpha)}$ is provable in $T_{\mathfrak{M}}$ then it has the proof where on the last step the rule (2) is applied to an axiom $(C \to D) \vdash (C \to D)^{(\alpha)}$.*

Theorem 4.4. (soundness). *Let $\mathfrak{M} =< G, f >$; $G = (V, E)$. If a sequence of sequents $\mathfrak{A} : A_1 \vdash B_1, \dots, A_n \vdash B_n$ is provable in $T_{\mathfrak{M}}$ then a set of requests $\mathfrak{Q} = ((A_1, B_1), \dots, (A_n, B_n))$ is successful for $\mathfrak{M}$.*

Proof*(sketch)* The theorem is proved by induction on the length of proof. If the length of proof is 0 then all $A_i \vdash B_i, i = 1 \dots n$ are axioms and it is easy to define an initialization for which an empty executionis successful for $\mathfrak{Q}$ on $\mathfrak{M}$.

Now let the theorem hold for all provable sequences of sequents with the lengths of proofs less than m. Let the proof of a sequence of sequents $\mathfrak{A} : A_1 \vdash B_1, \dots, A_n \vdash B_n$ have length m. Consider the last rule applied. *(a).* The last rule applied is rule (1). In this case, for some $i \in 1 \dots n$ the sequent $A_i \vdash B_i$ is $A_i'A_i'' \vdash F^{(\alpha)}CH$, for some $\alpha \in V$ and there exists such $B_i' \in Slist$ that the proof of $\mathfrak{A}$ has a form

$$\frac{A_1 \vdash B_1 \dots A_i' \vdash B_i'^{(\alpha)}C \;\; A_i'' \vdash (B_i' \to F)^{(\alpha)}H \dots A_n \vdash B_n}{A_1 \vdash B_1 \dots A_i'A_i'' \vdash F^{(\alpha)}CH \dots A_n \vdash B_n}(1)$$

The sequence of sequents

$$\mathfrak{A}' : A_1 \vdash B_1 \dots A_i' \vdash B_i'^{(\alpha)}C \qquad A_i'' \vdash (B_i' \to F)^{(\alpha)}H \dots A_n \vdash B_n$$

is provable and its proof has length $m - 1$. Then a set of requests ,

$$\begin{aligned}\mathfrak{Q}' &= ((A_1, B_1), \dots, (A_i', B_i'^{(\alpha)}C), (A_i'', (B_i' \to F)^{(\alpha)}H), \dots, (A_n, B_n)) \\ &= (Q_1', \dots, Q_i', Q_{i+1}', \dots, Q_{n+1}')\end{aligned}$$

is successful for $\mathfrak{M}$. Let f_{init}' be an initialization and $\sigma' = s_1 \dots s_l$ be a successful execution for $\mathfrak{Q}'$ on $\mathfrak{M}$. Then we construct a successful functioning for $\mathfrak{Q}$.

$$\text{Let } f_{init}(\alpha, r, j) = \begin{cases} f_{init}'(\alpha, r, j); & j < i \\ f_{init}'(\alpha, r, j) + f_{init}'(\alpha, r, j+1); & j = i \\ f_{init}'(\alpha, r, j+1); & j > i \end{cases}$$

Then the execution $\sigma = s_1, \dots, s_l, s_{l+1}$ where s_{l+1} is a resource transformation from B_i' and $(B_i' \to F)$ into F in node α is successful for $\mathfrak{Q}$ on $\mathfrak{M}$ with f_{init}.

The other three cases are considered in a similar way. Thus by induction the theorem is proved.

THEOREM 4.5. (completeness). *Let $\mathfrak{M} =< G, f >$; $G = (V, E)$. If a set of requests $\mathfrak{Q} = (Q_1, \dots, Q_n)$ is successful on $\mathfrak{M}$ then a sequence of sequents $\mathfrak{A} : A_1 \vdash B_1 \dots A_n \vdash B_n$ is provable in $T_{\mathfrak{M}}$.*

PROOF*(sketch)* The proof is made by induction on the length l of a succesful execution $\sigma = s_1, \dots, s_l$ for $\mathfrak{Q}$ on $\mathfrak{M}$. If $l = 0$ then for every $i \in 1 \dots n$, Q_i is $(A_{i1} \dots A_{ik_i}, A_{i1}^{(\alpha_{i1})} \dots A_{ik_i}^{(\alpha_{ik_i})}$ and an initial state C_{init} determined by initialization f_{init} is a final state C_{fin}. Let $A_{ij} = r_{1ij} \dots r_{t_{ij}ij}$. Then, for every $i \in 1 \dots n$, for every $j \in 1 \dots k_i$ and for every $h \in 1 \dots t_{ij}$ there exists a distinctive axiom $r_{hij} \vdash r_{hij}^{(\alpha_{ij})}$ in $T_{\mathfrak{M}}$. By applying rules of type (2)(*conj*) to these axioms we can obtain the sequence of sequents $\mathfrak{A}$.

Suppose now that for all sets of requests successful on $\mathfrak{M}$ with successful executions of length less than l the theorem is true. Consider a set of requests $\mathbf{\Omega}$ that has a successful execution $\sigma = s_1, \dots, s_l$ for some initialization f_{init}. Consider s_l – the last step of execution.

Case 1. s_l is transformation. Let $C_{init}(s_1, \dots, s_{l-1})C'$ and $C's_lC_{fin}$. Since s_l is transformation, for some $i \in 1 \dots n$ and $\alpha \in V$ in state C' $\{X, (X \rightarrow Y)\} \subseteq R'(\alpha, i)$ and in state C_{fin}. $R(\alpha, i) = (R'(\alpha, i) \cup \{Y\}) \backslash \{X, (X \rightarrow Y)\}$. An execution $\sigma' = s_1, \dots, s_{l-1}$ has length $l - 1$ and it is successful for a set of requests $\mathbf{\Omega}' = ((A_1, B_1), \dots, (A_i, X^{(\alpha)}(X \rightarrow Y)^{(\alpha)}D), \dots, (A_n, B_n))$, where B_i is equal to $Y^{(\alpha)}D$. By induction supposition a sequence of sequents $\mathfrak{A}'$: $A_1 \vdash B_1 \dots A_i \vdash (X(X \rightarrow Y))^{(\alpha)}D \dots A_n \vdash B_n$ is provable. By **lemma 4.1** the sequent $A_i \vdash (X(X - LY))^{(\alpha)}D$ is also provable. By **lemma 4.3** there exists a proof of this sequent which ends in the following way :

$$\frac{A_i' \vdash X^{(\alpha)}D \quad (X \rightarrow Y) \vdash (X \rightarrow Y)^{(\alpha)}}{A_i \vdash (X(X \rightarrow Y)^{(\alpha)}D}(2)$$

where A_i is equal to $A_i'(X \rightarrow Y)$.

Now we construct the proof of $A_i \vdash B_i$:

$$\frac{A_i' \vdash X^{(\alpha)}D \quad (X \rightarrow Y) \vdash (X \rightarrow Y)^{(\alpha)}}{A_i'(X \rightarrow Y) \vdash Y^{(\alpha)}D}(1)$$

Since in the proof of $A_i \vdash B_i$ we used the same axioms as in the proof of $A_i \vdash (X(X - LY))^{(\alpha)}D$, the sequence of sequents $\mathfrak{A}$: $A_1 \vdash B_1 \dots A_n \vdash B_n$ is provable.

Case 2. s_l is transmission. In this case rule (4) (*trans*) may be used to continue the proof.

Acknowledgements

Author is greatful to M.I.Dekhtyar for valuable comments on this paper. Author also wants to thank D.A. Arkhangelsky, D.O. Daderkin, I.Kh.Musikaev and M.A.Taitslin for useful discussions.

References

1. Arkhangelsky D.A., Dekhtyar M.I., Musikaev I.Kh., Taitslin M.A., *Linear Logic For Nets With Bounded Resources*, (submitted to LICS'94).
2. Archangelsky D.A., Taitslin M.A., *Linear Logic With Fixed Resources*, Annals of Pure and Appl. Math. Logic (to appear).
3. Girard J.Y., *Linear Logic*, Theoretical Computer Science 50 (1987), 1-102.
4. Lafont Y., *Interaction Nets,*, 17th ACM Symp. on Principles of Programming Languages (1990), San Francisco, Cal., 95-108.