Direct Encodings of NP-Complete Problems into Horn Sequents of Multiplicative Linear Logic

Satoshi Matsuoka

National Institute of Advanced Industrial Science and Technology (AIST),
1-1-1 Umezono, Tsukuba, Ibaraki, 305-8565 Japan
matsuoka@ni.aist.go.jp

In this paper, we provide direct encodings into Horn sequents of Multiplicative Linear Logic for two NP-complete problems, 3D MATCHING and PARTITION. Their correctness proofs are given by using a characterization of multiplicative proof nets.

1 Introduction

Around early 1990s, Max Kanovich introduced several Horn fragments of Linear Logic (see [5]). In particular, the multiplicative Horn fragment (for short HMLL) is a rather restricted subsystem of Intuitionistic Multiplicative Linear Logic. In [5], HMLL is shown to be NP-complete with regard to provability through the encoding of the 3-PARTITION problem ([2]). Moreover Krantz and Mogbil [7] gave another NP-completeness proof of HMLL through the encoding of the DIRECTED HAMILTONIAN CIRCUIT problem ([2]).

In this paper, we go forward further along this promising direction. We give the direct encoding of the 3D MATCHING problem as well as that of the PARTITION problem ([2]) into Horn sequents, which are sequents of HMLL. It is well-known that an NP-complete problem can also solve any other NP-complete problem through polynomial time transformations. But direct encodings are important because they provide more efficient solvers for NP-complete problems than indirect encodings through polynomial time transformations. In fact problem sizes encoded directly tend to be significantly smaller than those encoded by polynomial time transformations from the viewpoint of a practical level, although both are related by polynomials.

To solve NP-completeness problems is important: to obtain more optimized solutions of practical optimization problems means to reduce tangible resources more drastically for systems constructed based on these solutions. Recent progress of SAT solvers has enabled this at a practical level (for example, see [8]). We would like to address this topic from another logical point of view, i.e, from Linear Logic. In this approach we exploit *provability* of Linear Logic instead of *satisfiability* of classical logic. Although combinatorial NP-complete problems can be encoded into SAT, such encodings are usually complicated and difficult for humans to understand. On the other hand, the four direct encodings into Horn sequents mentioned above are surprisingly simple and easy to understand. The existence of such natural direct encodings seems to suggest that more complicated practical combinatorial problems can be solved using a Linear Logic proof search engine efficiently instead of using a SAT solver.

In fact the software called *Proof Net Calculator* [9], which we have developed, includes an implementation of the 3D MATCHING problem as well as that of the DIRECTED HAMILTONIAN CIRCUIT problem, using the encodings mentioned above. Although we have not exploited *problem-specific* optimizations

Submitted to:
HCVS 2017

yet, Proof Net Calculator can solve instances of these problems which average persons (like the author) seem difficult to solve, where we utilize *encoding-specific* optimizations using *ID-links dependency relations*. The technical details will be given elsewhere.

2 Intuitionistic Multiplicative Linear Logic, Horn sequents, and MLL

2.1 Intuitionistic Multiplicative Linear Logic and Horn sequents

In this section we introduce the system of Intuitionistic Multiplicative Linear Logic (for short IMLL) and then Horn sequents in IMLL. Two NP-complete problems, `3D MATCHING` and `PARTITION` are encoded as Horn sequents as well as `3PARTITION` in [5] and `DIRECTED HAMILTONIAN CIRCUITS` in [7]. Our terminology and notation with regard to sequent calculi are standard, although we exclude the cut rule, since we only deal with cut-free systems in this paper. For more technical details, for example, see [4]. We do not deal with the multiplicative Horn fragment of Linear Logic (for short HMLL) in [5]. HMLL includes the cut rule in an essential way: the cut elimination theorem does not hold in HMLL. All we need in this paper are cut-free systems.

We denote *atomic formulas* by $p,q,r,\ldots$. Then we define *IMLL formulas*, which are denoted by $A,B,C,\ldots$, by the following grammar:

$$A ::= p \mid A \multimap B \mid A \otimes B$$

We denote *multisets of IMLL formulas* by $\Sigma, \Sigma_1, \Sigma_2, \ldots$. An IMLL sequent is a pair (Σ, A). We write an IMLL sequent (Σ, A) as $\Sigma \vdash A$. The inference rules of IMLL are as follows:

$$\mathrm{I} \quad \frac{}{A \vdash A}$$

$$\mathrm{L}\multimap \quad \frac{\Sigma_1 \vdash A \qquad B, \Sigma_2 \vdash C}{\Sigma_1, A \multimap B, \Sigma_2 \vdash C} \qquad \mathrm{R}\multimap \quad \frac{\Sigma, A \vdash B}{\Sigma \vdash A \multimap B}$$

$$\mathrm{L}\otimes \quad \frac{\Sigma, A, B \vdash C}{\Sigma, A \otimes B \vdash C} \qquad \mathrm{R}\otimes \quad \frac{\Sigma_1 \vdash A \qquad \Sigma_2 \vdash B}{\Sigma_1, \Sigma_2 \vdash A \otimes B}$$

By a *simple formula* we mean a formula that consists of only atomic formulas and the $\otimes$ connective. We denote simple formulas by $X,Y,Z,\ldots$. By a *Horn implication* we mean a formula that has the form $X \multimap Y$. A *Horn sequent* is an IMLL sequent that has the form $X, \Sigma \vdash Y$, where each formula in Σ is a Horn implication. We note that the Horn sequents are a rather restricted class of the IMLL sequents.

2.2 Multiplicative Linear Logic

Next we introduce the system of Multiplicative Linear Logic (for short MLL). We define *MLL formulas*, which are denoted by $F,G,H,\ldots$, by the following grammar:

$$F ::= p \mid p^{\perp} \mid F \otimes G \mid F \parr G$$

The negation of F, which is denoted by $F^{\perp}$ is defined as follows:

$$\begin{array}{lcl} (p)^{\perp} & = & p^{\perp} \\ (F \otimes G)^{\perp} & = & G^{\perp} \parr F^{\perp} \\ (F \parr G)^{\perp} & = & G^{\perp} \otimes F^{\perp} \end{array}$$

We denote *multisets of MLL formulas* by $\Lambda, \Lambda_1, \Lambda_2, \ldots$. An MLL sequent is a multiset of MLL formulas Λ. We write an MLL sequent Λ as $\vdash \Lambda$. The inference rules of MLL are as follows:

$$\text{ID} \quad \frac{}{\vdash F^{\perp}, F}$$

$$\otimes \quad \frac{\vdash \Lambda_1, F \qquad \vdash \Lambda_2, G}{\vdash \Lambda_1, \Lambda_2, F \otimes G} \qquad \invamp \quad \frac{\vdash \Lambda, F, G}{\vdash \Lambda, F \invamp G}$$

We define a translation from IMLL sequents to MLL sequents by

$$A_1, \ldots, A_n \vdash B \quad \mapsto \quad \vdash (A_1)^-, \ldots, (A_n)^-, B^+$$

where $(-)^-$ and $(-)^+$ are defined inductively as follows:

$$\begin{array}{lclclcl}
(p)^- & = & p^{\perp} & & (p)^+ & = & p \\
(A \multimap B)^- & = & (B)^- \otimes (A)^+ & & (A \multimap B)^+ & = & (A)^- \invamp (B)^+ \\
(A \otimes B)^- & = & (B)^- \invamp (A)^- & & (A \otimes B)^+ & = & (A)^+ \otimes (B)^+
\end{array}$$

Proposition 2.1 *A sequent $A_1, \ldots, A_n \vdash B$ is provable in IMLL if and only if $\vdash (A_1)^-, \ldots, (A_n)^-, B^+$ is provable in MLL.*

Proof: For example, see [10]. □

By Proposition 2.1 we can discuss Horn sequent encodings of NP-complete problems in the framework of MLL.

2.3 MLL proof nets

Next we introduce MLL proof nets. We use MLL proof nets in order to prove the correctness of our encodings into Horn sequents.

Figure 1 shows the *MLL links* we use. Each MLL link has a few MLL formulas. Such an MLL formula is a conclusion or a premise of the MLL link, which is specified as follows:

1. In an ID-link, each of F and $F^{\perp}$ is called a conclusion of the link;
2. In an $\otimes$-link, each of F and G is called a premise of the link and $F \otimes G$ is called a conclusion of the link;
3. In an $\invamp$-link, each of F and G is called a premise of the link and $F \invamp G$ is called a conclusion of the link.

An MLL *proof structure* Θ is a set of MLL links that satisfies the following conditions:

1. For each link L in Θ, each conclusion of L can be a premise of at most one link other than L in Θ;
2. For each link L in Θ, each premise of L must be a conclusion of exactly one link other than L in Θ.

An MLL *proof net* is an MLL proof structure that is constructed by the rules in Figure 2. Note that each rule in Figure 2 has the corresponding inference rule in the MLL sequent calculus. Any MLL proof structure is not necessarily an MLL proof net.

Next we introduce a characterization of MLL proof nets using the notion of DR-switchings. A DR-switching S for an MLL proof structure Θ is a function from the set of $\invamp$-links in Θ to $\{0,1\}$. The DR-graph $S(\Theta)$ for Θ and S is defined by the rules of Figure 3. Then the following characterization holds.

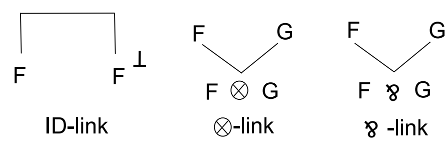

Figure 1: MLL Links

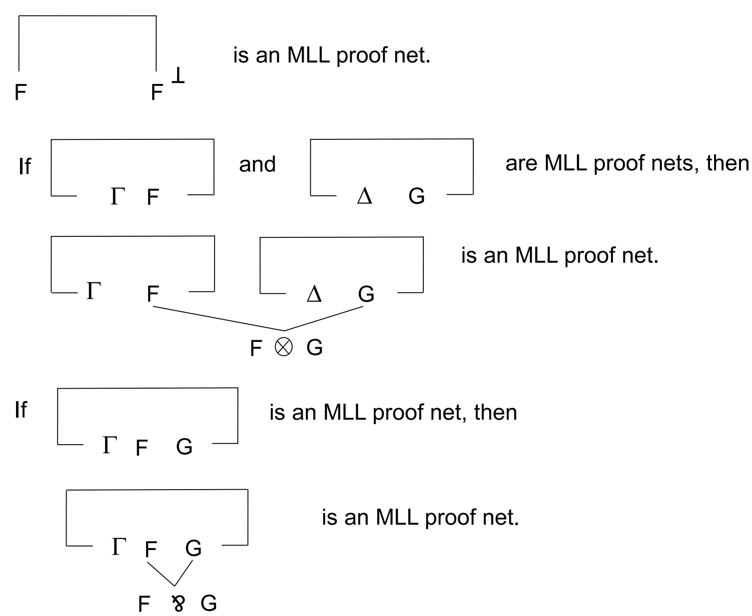

Figure 2: Definition of MLL Proof Nets

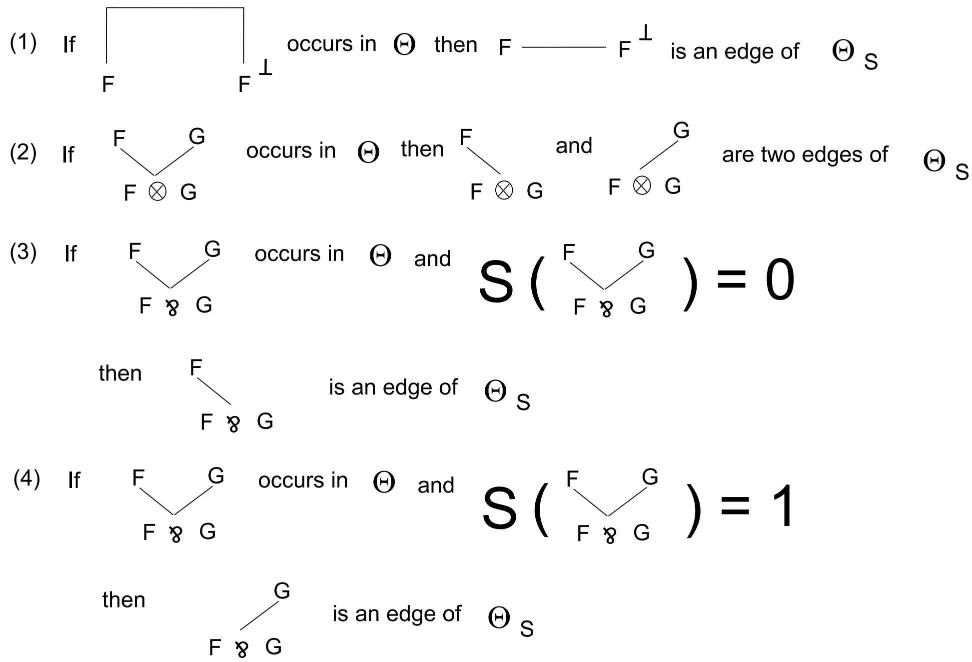

Figure 3: Definition of DR graphs

Theorem 2.1 ([1]) *An MLL proof structure Θ is an MLL proof net if and only if for any DR-switching S for Θ, the DR-graph S_Θ is acyclic and connected.*

Next we introduce a few notions for MLL proof search based on MLL proof nets. An MLL *proof forest* Θ_0 is a set of MLL links obtained from an MLL proof structure Θ by deleting all ID-links in Θ. An *ID-links set* π for an MLL proof forest Θ_0 is a set of ID-links such that $\Theta_0 \cup \pi$ is an MLL proof structure. Then we write $\Theta_0 \cup \pi$ as Θ_0^π. We say that an MLL proof forest Θ_0 has an MLL proof net if there is an ID-links set π for Θ_0 such that Θ_0^π has an MLL proof net. An MLL formula A is a conclusion of an MLL proof forest Θ_0 if there is a link L in Θ_0 such that A is a conclusion of L and there is no link L' in Θ_0 such that A is a premise of L'.

Proposition 2.2 *Let $A_1, \ldots, A_n \vdash B$ be an IMLL sequent and $(A_1)^-, \ldots, (A_n)^-, B^+$ be the conclusions of an MLL forest Θ_0. Then $A_1, \ldots, A_n \vdash B$ is provable in IMLL if and only if Θ_0 has an MLL proof net.*

Proof: By Proposition 2.1, $A_1, \ldots, A_n \vdash B$ is provable in IMLL if and only if $\vdash (A_1)^-, \ldots, (A_n)^-, B^+$ is provable in MLL. Moreover, $\vdash (A_1)^-, \ldots, (A_n)^-, B^+$ is provable in MLL if and only if the MLL proof forest with the conclusions $(A_1)^-, \ldots, (A_n)^-, B^+$ has an MLL proof net. $\square$

3 The Encoding of `3D MATCHING`

3.1 Preliminaries

Notation 1 *Let S be a set. Let L be the set of all lists over S such that each list in L has the same length. Then we define an equivalence relation* Perm *over L as follows: (ℓ_1, ℓ_2) is in* Perm *if for each $s \in S$, the number of occurrences of s in ℓ_1 is the same as that of ℓ_2. Each equivalence class over* Perm *is a multiset. When ℓ is a list over S, let $\ell/$*Perm *be the multiset that includes ℓ.*

Definition 3.1 (3D MATCHING **[6]**) *Let A, B, C be finite sets such that $|A| = |B| = |C| = n$.* 3D MATCHING *is the problem that when a given set $T \subseteq A \times B \times C$, decides whether or not there is a subset T_0 of T such that $|T_0| = n$ and*

$$\begin{array}{rcl} A &=& \{a \in A \,|\, \exists b \in B. \exists c \in C. \langle a,b,c\rangle \in T_0\} \\ B &=& \{b \in B \,|\, \exists a \in A. \exists c \in C. \langle a,b,c\rangle \in T_0\} \\ C &=& \{c \in C \,|\, \exists a \in A. \exists b \in b. \langle a,b,c\rangle \in T_0\}\ . \end{array}$$

We suppose $|T| = n+m$, where $m \geq 0$. We assume that the elements of T are ordered and the list is

$$\langle a_{i_1}, b_{j_1}, c_{k_1}\rangle, \ldots, \langle a_{i_n}, b_{j_n}, c_{k_n}\rangle, \langle a_{i_{n+1}}, b_{j_{n+1}}, c_{k_{n+1}}\rangle, \ldots, \langle a_{i_{n+m}}, b_{j_{n+m}}, c_{k_{n+m}}\rangle\ .$$

Then we define three *multisets* A_T, B_T, C_T by

$$\begin{array}{rcl} A_T &=& \langle a_{i_1}, \ldots, a_{i_n}, a_{i_{n+1}}, \ldots, a_{i_{n+m}}\rangle/\mathrm{Perm} \\ B_T &=& \langle b_{j_1}, \ldots, b_{j_n}, b_{j_{n+1}}, \ldots, b_{j_{n+m}}\rangle/\mathrm{Perm} \\ C_T &=& \langle c_{k_1}, \ldots, c_{k_n}, c_{k_{n+1}}, \ldots, c_{k_{n+m}}\rangle/\mathrm{Perm} \end{array}$$

Without loss of generality, we suppose the following conditions:

1. For each $a \in A$, there is $\ell\,(1 \leq \ell \leq n+m)$ such that $a = a_{i_\ell}$;
2. For each $b \in B$, there is $\ell\,(1 \leq \ell \leq n+m)$ such that $b = b_{j_\ell}$;
3. For each $c \in C$, there is $\ell\,(1 \leq \ell \leq n+m)$ such that $c = c_{k_\ell}$.

Otherwise, we can determine that this instance has no solution. Moreover we define three *multisets* $A_{\mathrm{co}}, B_{\mathrm{co}}, C_{\mathrm{co}}$ by

$$\begin{array}{rcl} A_{\mathrm{co}} &=& A_T - A_{\mathrm{mul}} \\ B_{\mathrm{co}} &=& B_T - B_{\mathrm{mul}} \\ C_{\mathrm{co}} &=& C_T - C_{\mathrm{mul}} \end{array}$$

where $A_{\mathrm{mul}}, B_{\mathrm{mul}}, C_{\mathrm{mul}}$ are multisets that have the same elements as A, B, C respectively such that each element occurs exactly once in $A_{\mathrm{mul}}, B_{\mathrm{mul}}, C_{\mathrm{mul}}$ respectively. So, $|A_{\mathrm{mul}}| = |B_{\mathrm{mul}}| = |C_{\mathrm{mul}}| = n$. Then without loss of generality, we can describe as

$$\begin{array}{rclcrcl} A_{\mathrm{mul}} &=& \langle a_1, \ldots, a_n\rangle/\mathrm{Perm} && A_{\mathrm{co}} &=& \langle a_{i'_1}, \ldots, a_{i'_m}\rangle/\mathrm{Perm} \\ B_{\mathrm{mul}} &=& \langle b_1, \ldots, b_n\rangle/\mathrm{Perm} && B_{\mathrm{co}} &=& \langle b_{j'_1}, \ldots, b_{j'_m}\rangle/\mathrm{Perm} \\ C_{\mathrm{mul}} &=& \langle c_1, \ldots, c_n\rangle/\mathrm{Perm} && C_{\mathrm{co}} &=& \langle c_{k'_1}, \ldots, c_{k'_m}\rangle/\mathrm{Perm} \end{array}$$

3.2 The encoding into a Horn sequent

In this section we give our encoding of 3D MATCHING problem into a Horn sequent. We need a few auxiliary formulas. For each $\ell\,(1 \leq \ell \leq n+m)$, we define

$$FT_\ell \quad = \quad (b_{j_\ell} \otimes c_{k_\ell}) \multimap a_{i_\ell}$$

Moreover we define

$$\begin{array}{rcl} FA_{\mathrm{co}} & = & a_{i'_1} \otimes \cdots \otimes a_{i'_m} \\ FB_{\mathrm{co}} & = & b_{i'_1} \otimes \cdots \otimes b_{j'_m} \\ FC_{\mathrm{co}} & = & c_{k'_1} \otimes \cdots \otimes b_{k'_m} \end{array}$$

Finally we define

$$F_{\mathrm{I}} \quad = \quad FA_{\mathrm{co}} \multimap ((b_1 \otimes \cdots \otimes b_n) \otimes (c_1 \otimes \cdots \otimes c_n))$$

Then we define a sequent as

$$\Gamma_{\texttt{3DMATCHING}} \quad = \quad FB_{\mathrm{co}} \otimes FC_{\mathrm{co}}, F_{\mathrm{I}}, FT_1, \ldots, FT_n, FT_{n+1}, \ldots, FT_{n+m} \vdash a_1 \otimes \cdots \otimes a_n$$

It is obvious that $\Gamma_{\texttt{3DMATCHING}}$ is a Horn sequent and the encoding is a polynomial reduction.

3.3 The correctness proof

In order to prove the correctness of the encoding, we exploit the characterization of MLL proof nets (Theorem 2.1). We construct an MLL proof forest, which corresponds to the Horn sequent $\Gamma_{\texttt{3DMATCHING}}$.

For each $\ell\,(1 \le \ell \le n+m)$, the forest Θ_ℓ corresponding to FT_ℓ is shown in Figure 4. The forest Θ_{I}

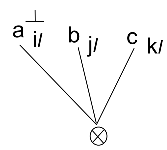

Figure 4: Triple device Θ_ℓ

corresponding to F_{I} is shown in Figure 5. Then the forest Θ_{F} corresponding to the rest in $\Gamma_{\texttt{3DMATCHING}}$ is

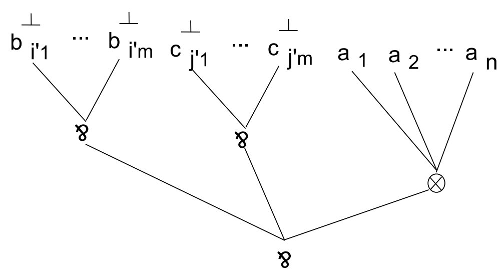

Figure 5: I-device Θ_{I}

shown in Figure 6. Then we define an MLL proof forest Θ_0 as

$$\Theta_0 \quad = \quad \Theta_{\mathrm{I}} \cup \bigcup_{1 \le \ell \le n+m} \Theta_\ell \cup \Theta_{\mathrm{F}}$$

Then if we note that $\vdash \Delta, A \parr B$ is provable in MLL if and only if $\vdash \Delta, A, B$ is provable in MLL, then we can easily see that $\Gamma_{\texttt{3DMATCHING}}$ is provable in IMLL if and only if Θ_0 has an MLL proof net by Proposition 2.2.

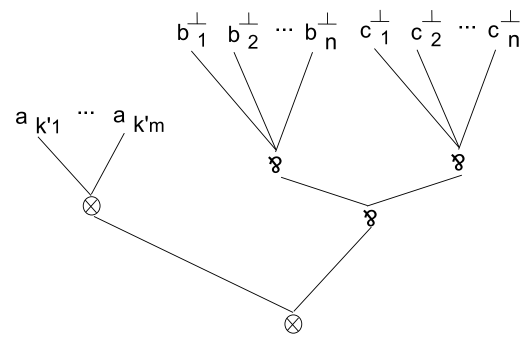

Figure 6: F-device Θ_{F}

Theorem 3.1 *An instance of* 3D MATCHING *has a solution if and only if there is an ID-links set π for the corresponding MLL proof forest Θ_0 such that Θ_0^π is an MLL proof net.*

Proof: We assume that we have a solution $T_0 \subseteq A \times B \times C$. Then without loss of generality we can write as

$$T_0 = \{\langle a_{i_1}, b_{j_1}, c_{k_1}\rangle, \ldots, \langle a_{i_n}, b_{j_n}, c_{k_n}\rangle\}\ .$$

Moreover we have

$$\begin{array}{rcl} A & = & \{a_{i_1}, \ldots, a_{i_n}\} \\ B & = & \{b_{j_1}, \ldots, b_{j_n}\} \\ C & = & \{c_{k_1}, \ldots, c_{k_n}\}\ . \end{array}$$

Then for each $\ell\,(1 \leq \ell \leq n)$, there is exactly one triple device Θ_{q_ℓ} corresponding to $\langle a_{i_\ell}, b_{j_\ell}, c_{k_\ell}\rangle$ in T_0 where $1 \leq q_\ell \leq n+m$. Let

$$\mathscr{T}_0 = \{\Theta_{q_1}, \ldots, \Theta_{q_n}\}$$

Moreover we can write the set of the triple devices that do not appear in $\mathscr{T}_0$ as

$$\mathscr{T}_1 = \{\Theta_{q_{n+1}}, \ldots, \Theta_{q_{n+m}}\}$$

We construct an ID-links set π for Θ_0 as follows:

(1) For each $\ell\,(1 \leq \ell \leq n)$ π includes the ID-link that connects the literal $a_{i_\ell}^\perp$ in the triple device Θ_{q_ℓ} to a_{i_ℓ} in the I-device Θ_{I}.

(2) For each $\ell\,(1 \leq \ell \leq n)$ π includes the ID-link that connects the literal b_{j_ℓ} in the triple device Θ_{q_ℓ} to $b_{j_\ell}^\perp$ in the F-device Θ_{F}.

(3) For each $\ell\,(1 \leq \ell \leq n)$ π includes the ID-link in π that connects the literal c_{k_ℓ} in the triple device Θ_{q_ℓ} to $c_{k_\ell}^\perp$ in the F-device Θ_{F}.

(4) For each $\ell\,(n+1 \leq \ell \leq n+m)$ π includes an ID-link that connects the literal $a_{i_\ell}^\perp$ in the triple device Θ_{q_ℓ} to a literal a_{i_ℓ} in the F-device Θ_{F}.

(5) For each $\ell\,(n+1 \leq \ell \leq n+m)$ π includes an ID-link that connects the literal b_{j_ℓ} in the triple device Θ_{q_ℓ} to a literal $b_{j_\ell}^\perp$ in the I-device Θ_{I}.

(6) For each $\ell\,(n+1 \le \ell \le n+m)$ π includes an ID-link that connects the literal c_{k_ℓ} in the triple device Θ_{q_ℓ} to a literal $c_{k_\ell}^{\perp}$ in the I-device Θ_{I}.

Then we can easily see that Θ_0^{π} is an MLL proof net.
Conversely we assume that we do not have any solution $T_0 \subseteq A \times B \times C$. In this case we can not find the ID-links set π for Θ_0 described above. So any ID-links set π for Θ_0 must have the following property: There is some $\ell\,(1 \le \ell \le n+m)$ such that

(1) the literal $a_{i_\ell}^{\perp}$ in Θ_ℓ connects to a_{i_ℓ} of the F-device Θ_{F} in π, and

(2) the literal b_{j_ℓ} in Θ_ℓ connects to $b_{j_\ell}^{\perp}$ of the F-device Θ_{F} in π or

(2') the literal c_{k_ℓ} in Θ_ℓ connects to $c_{k_\ell}^{\perp}$ of the F-device Θ_{F} in π.

Then we can find a DR-switching S for Θ_0^{π} such that the DR-graph $S(\Theta_0^{\pi})$ has a cycle that passes through $a_{i_\ell}^{\perp}, a_{i_\ell}$ and $b_{j_\ell}^{\perp}, b_{j_\ell}$, or $a_{i_\ell}^{\perp}, a_{i_\ell}$ and $c_{k_\ell}^{\perp}, c_{k_\ell}$. □

Corollary 3.1 *An instance of* `3D MATCHING` *has a solution if and only if the sequent* $\Gamma_{\mathtt{3DMATCHING}}$ *for the instance is provable in MLL.*

We note that this result can be easily extended to the n-D `MATCHING` problem for any $n\,(n \ge 2)$.

4 The Encoding of `PARTITION`

4.1 Preliminaries

Definition 4.1 (`PARTITION` [6]) *Let A be a finite set and s be a function from A to $\mathbb{Z}^{+}$.* `PARTITION` *is the problem that decide whether or not there is a subset $A' \subseteq A$ such that*

$$\sum_{s \in A'} s(a) = \sum_{a \in A - A'} s(a)\ .$$

The problem is different from `3-PARTITION` used in [5] and an NP-complete problem ([2]). In particular our encoding below cannot be derived from the `3-PARTITION` encoding in [5] directly.

We assume that the elements of A are ordered and the list is

$$a_1, a_2, \ldots, a_k$$

Let

$$t = \sum_{1 \le i \le k} s(a_i)\ .$$

Note that for any subset $A' \subseteq A$,

$$t = \sum_{s \in A'} s(a) + \sum_{a \in A - A'} s(a)\ .$$

If A' is a solution, then the following equation must hold:

$$t = 2 \sum_{s \in A'} s(a)\ .$$

Then t must be even. So without loss of generality, we can assume t is even.

4.2 The encoding into a Horn sequent

In this section we give our encoding of the PARTITION problem into a Horn sequent. We need a few auxiliary formulas.

For each $i\,(1 \leq i \leq k)$, we define $F_{\text{one}i}$ and $F_{\text{ano}i}$ as

$$\begin{array}{rcl} F_{\text{one}i} & = & a_i \multimap \overbrace{b^\perp \otimes b^\perp \otimes \cdots b^\perp}^{s(a_i)} \\ F_{\text{ano}i} & = & a_i \multimap \overbrace{c^\perp \otimes c^\perp \otimes \cdots c^\perp}^{s(a_i)} \end{array}$$

Let F_{bc} be

$$F_{bc} \quad = \quad (\overbrace{b \otimes b \otimes \cdots \otimes b}^{t/2}) \otimes (\overbrace{c \otimes c \otimes \cdots \otimes c}^{t/2})$$

We define F_{1st} and F_{2nd} as

$$\begin{array}{rcl} F_{\text{1st}} & = & F_{\text{bc}} \multimap a_1 \otimes a_2 \otimes \cdots \otimes a_k \\ F_{\text{2nd}} & = & F_{\text{bc}} \multimap e \end{array}$$

Then we define a sequent as

$$\Gamma_{\text{PARTITION}} \quad = \quad a_1 \otimes a_2 \otimes \cdots \otimes a_k, F_{\text{one}1}, \ldots, F_{\text{one}k}, F_{\text{ano}1}, \ldots, F_{\text{ano}k}, F_{\text{1st}}, F_{\text{2nd}} \vdash e$$

It is obvious that $\Gamma_{\text{PARTITION}}$ is a Horn sequent and the encoding is a polynomial reduction.

We give an informal meaning for these formulas as follows:

- The formula $a_1 \otimes a_2 \otimes \cdots \otimes a_k$ gives the multiset of all items.
- The formula $F_{\text{one}i}$ gives the weight for the item a_i.
- The formula $F_{\text{ano}i}$ also gives the weight for the item a_i.
- The role of F_{1st} is a balance. If we are able to partition the set A of the items into two disjoint sets A' and $A - A'$ such that the sum of weights of A' is equal to that $A - A'$, then again we get the multiset of all items $a_1 \otimes a_2 \otimes \cdots \otimes a_k$.
- The role of F_{2nd} is also a balance. If we succeed in the partition mentioned above, then we get the final formula e.

4.3 The correctness proof

In order to prove the correctness of the encoding, we exploit the characterization of MLL proof nets (Theorem 2.1). We construct an MLL proof forest, which corresponds to the Horn sequent $\Gamma_{\text{PARTITION}}$.

Let the MLL forest shown in Figure 7 be Θ_{I}. For each $i\,(1 \leq i \leq k)$, let the MLL forest shown in Figure 8 be $\Theta_{\text{one}i}$. For each $i\,(1 \leq i \leq k)$, let the MLL forest shown in Figure 9 be $\Theta_{\text{ano}i}$. Let the MLL forest shown in Figure 10 be Θ_{1st}. Let the MLL forest shown in Figure 11 be Θ_{2nd}. Finally let the MLL forest shown in Figure 12 be Θ_{F}. Then we define an MLL proof forest Θ_0 as

$$\Theta_0 = \Theta_{\text{I}} \cup \bigcup_{1 \leq i \leq k} \Theta_{\text{one}i} \cup \bigcup_{1 \leq i \leq k} \Theta_{\text{ano}i} \cup \Theta_{\text{1st}} \cup \Theta_{\text{2nd}} \cup \Theta_{\text{F}}$$

Then by Proposition 2.2 $\Gamma_{\texttt{PARTITION}}$ is provable in IMLL if and only if Θ_0 has an MLL proof net.

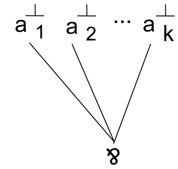

Figure 7: I-device Θ_{I}

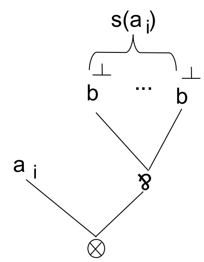

Figure 8: One Side device $\Theta_{\mathrm{one}i}$

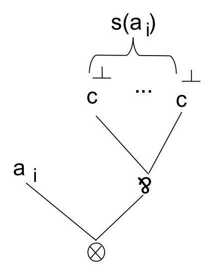

Figure 9: Another Side device $\Theta_{\mathrm{ano}i}$

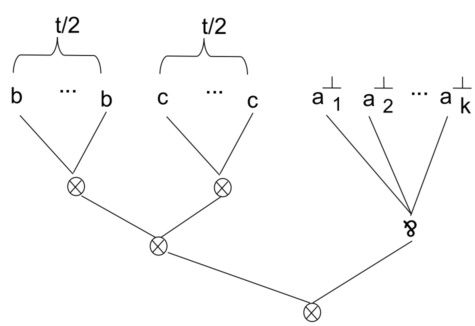

Figure 10: The First Matching Device Θ_{1st}

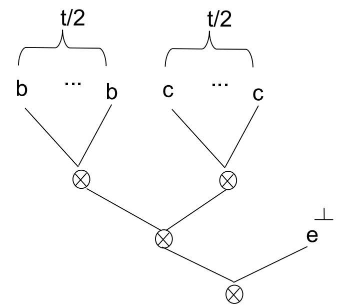

Figure 11: The Second Matching Device Θ_{2nd}

Figure 12: F-Device Θ_{F}

Theorem 4.1 *An instance of* `PARTITION` *has a solution if and only if there is an ID-links set π for the corresponding MLL proof forest Θ_0 such that Θ_0^π is an MLL proof net.*

Proof: We assume that we have a solution $A' \subseteq A$. Then without loss of generality we can write as

$$\begin{array}{rcl} A' & = & \{a_1, \ldots, a_{k_0}\} \\ A - A' & = & \{a_{k_0+1}, \ldots, a_k\} \end{array}$$

Since A' is a solution, as mentioned above, we can write as

$$t/2 = \sum_{s \in A'} s(a) = \sum_{s \in A - A'} s(a) \; .$$

Then we construct an ID-links set π for Θ_0 as follows:

(1) For each $i \, (1 \leq i \leq k_0)$, $a_i^\perp$ in Θ_{I} is connected to a_i in $\Theta_{\text{one}i}$ and each of $b^\perp$ in $\Theta_{\text{one}i}$ is connected to b in Θ_{1st}.

(2) For each $i \, (1 \leq i \leq k_0)$, $a_i^\perp$ in $\Theta_{\text{1st}i}$ is connected to a_i in $\Theta_{\text{ano}i}$ and each of $c^\perp$ in $\Theta_{\text{ano}i}$ is connected to c in Θ_{2nd}.

(3) For each $i \, (k_0 + 1 \leq i \leq k)$, $a_i^\perp$ in Θ_{I} is connected to a_i in $\Theta_{\text{ano}i}$ and each of $c^\perp$ in $\Theta_{\text{ano}i}$ is connected to c in Θ_{1st}.

(4) For each $i \, (k_0 + 1 \leq i \leq k)$, $a_i^\perp$ in Θ_{1st} is connected to a_i in $\Theta_{\text{one}i}$ and each of $b^\perp$ in $\Theta_{\text{one}i}$ is connected to b in Θ_{2nd}.

(5) The literal $e^\perp$ in Θ_{2nd} is connected to e in Θ_{F}.

It is obvious that Θ_0^π is an MLL proof net.
Conversely, we assume that we do not have any solution $A' \subseteq A$. This means that for any subset $A' \subseteq A$,

$$t/2 \neq \sum_{s \in A'} s(a) \neq \sum_{s \in A - A'} s(a) \neq t/2 \; .$$

Then any ID-links set π for Θ_0 must have the following property:

There is some $i\,(1 \leq i \leq k)$ such that

(1) the literal $a_i^{\perp}$ in Θ_{1st} is connected to the literal a_i in $\Theta_{\mathrm{one}i}$;

(2) a literal b in Θ_{1st} is connected to a literal $b^{\perp}$ in $\Theta_{\mathrm{one}i}$,

or

(1') the literal $a_i^{\perp}$ in Θ_{1st} is connected to the literal a_i in $\Theta_{\mathrm{ano}i}$;

(2') a literal c in Θ_{1st} is connected to a literal $c^{\perp}$ in $\Theta_{\mathrm{ano}i}$.

Then there is a DR-switching S for Θ_0^{π} such that $S(\Theta_0^{\pi})$ has a cycle including all literals mentioned in (1) and (2), or (1') and (2'). $\square$

Corollary 4.1 *An instance of* `PARTITION` *has a solution if and only if the sequent* $\Gamma_{\mathrm{PARTITION}}$ *for the instance is provable in MLL.*

5 Concluding Remarks

In this paper we showed that Horn sequents of Linear Logic are extremely useful for formalizing combinatorial NP-completeness problems. This suggests that more complicated practical combinatorial NP-complete problems can be directly encoded into these sequents. Although Horn sequents of Linear Logic has not received much attention, we believe that the research direction is promising.

In addition, The `3SAT` problem can be encoded into the `3D MATCHING` problem in a standard way (for example, see [2]). So we can obtain a `SAT` solver through MLL proof search in this way.

References

[1] V. Danos & R. Regnier (1989): *The structure of multiplicatives*. Archive for Mathematical Logic 28, pp. 181–203, doi:10.1007/BF01622878.

[2] M. R. Garey & D. S. Johnson (1979): *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company.

[3] J.-Y. Girard (1987): *Linear Logic*. Theoretical Computer Science 50, pp. 1–102, doi:10.1016/0304-3975(87)90045-4.

[4] J.-Y. Girard, Y. Lafont & P. Taylor (1989): *Proofs and Types*. Cambridge University Press.

[5] M. I. Kanovich (1994): *The complexity of Horn fragments of Linear Logic*. Annals of Pure and Applied Logic 69, pp. 195–241, doi:10.1016/0168-0072(94)90085-X.

[6] R. M. Karp (1972): *Reducibility among Combinatorial Problems*. In Raymond E. Miller, James W. Thatcher & Jean D. Bohlinger, editors: Complexity of Computer Computations: Proceedings of a symposium on the Complexity of Computer Computations, Springer US, Boston, MA, pp. 85–103, doi:10.1007/978-1-4684-2001-2_9.

[7] T. Krantz & V. Mogbil (2001): *Encoding Hamiltonian circuits into multiplicative linear logic*. Theoretical Computer Science 266, pp. 987–996, doi:10.1016/S0304-3975(00)00381-9.

[8] S. Malik & L. Zhang (2009): *Boolean Satisfiability: From Theoretical Hardness to Practical Success*. Communication of the ACM 52, pp. 76–82, doi:10.1145/1536616.1536637.

[9] Satoshi Matsuoka: *Proof Net Calculator*. Available at `https://staff.aist.go.jp/s-matsuoka/PNCalculator/index.html`.

[10] A. M. Murawski & C.-H. L. Ong (2006): *Fast Verification of MLL Proof Nets via IMLL*. ACM Transaction on Computational Logic 7, pp. 473–498, doi:10.1145/1149114.1149116.

[11] P. J. de Naurois & V. Mogbil (2011): *Correctness of linear logic proof structures is NL-complete*. *Theoretical Computer Science* 412, pp. 1941–1957, doi:10.1016/j.tcs.2010.12.020.