Multiple Conclusion Intuitionistic Linear Logic and Cut Elimination

Harley Eades III and Valeria de Paiva

February 17, 2015

Abstract

Full Intuitionistic Linear Logic (FILL) was first introduced by Hyland and de Paiva as one of the results of their investigation into a categorical understanding of Gödel's Dialectica interpretation. FILL went against current beliefs that it was not possible to incorporate all of the linear connectives, e.g. tensor, par, and implication, into an intuitionistic linear logic. They showed that it is natural to support all of the connectives given sequents that have multiple hypotheses and multiple conclusions. To enforce intuitionism de Paiva's original formalization of FILL used the well-known Dragalin restriction, forcing the implication right rule to have only a single conclusion in its premise, but Schellinx showed that this results in a failure of cut-elimination. To overcome this failure Hyland and de Paiva introduced a term assignment for FILL that eliminated the need for the strong restriction. The main idea was to first relax the restriction by assigning variables to each hypothesis and terms to each conclusion. Then when introducing an implication on the right enforcing that the variable annotating the hypothesis being discharged is only free in the term annotating the conclusion of the implication. Bierman showed in a short note that this formalization of FILL still did not enjoy cut-elimination, because of a flaw in the left rule for par. However, Bellin proposed an alternate left rule for par and conjectured that by adopting his rule cut-elimination is restored. In this note we show that adopting Bellin's proposed rule one does obtain cut-elimination for FILL, as suggested. Additionally, we show that this new formalization can be modeled by a new form of dialectica category called order-enriched dialectica category, and discuss future work giving FILL a semantics in terms of Lorenzen games.

1 Introduction

A commonly held belief during the early history of linear logic was that the linear-connective par could not be incorporated into an intuitionistic linear logic. This belief was challenged when de Paiva gave a categorical understanding of Gödel's Dialectica interpretation in terms of dialectica categories [2]. Upon setting out on her investigation she initially believed that dialectica categories would end up being a model of intuitionistic logic, but to her surprise they are actually models of intuitionistic linear logic, containing the linear connectives: tensor, par, implication, and their units. Furthermore, unlike other models at that time the units did not collapse into a single object.

Armed with this semantic insight de Paiva gave the first formalization of Full Intuitionistic Linear Logic (FILL) [?]. FILL is a sequent calculus with multiple conclusion in addition to multiple hypotheses. Intuitionism was enforced by using the well-known Dragalin restriction. The classical implication right rule has the form:

$$\frac{\Gamma, A \vdash B, \Delta}{\Gamma \vdash A \multimap B, \Delta}\ \textsc{impR}$$

The Dragalin restriction is placed on the premise of the implication right rule enforcing that there is only a single conclusion in the premise resulting in the rule:

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \multimap B}\ \textsc{impR}$$

All the other rules are left unrestricted. The unfortunate consequence of this restriction is that it breaks cut-elimination [7].

Later, Hyland and de Paiva gave an alternate formalization of FILL in the hopes to regain cut-elimination [5]. This new formalization lifted the Dragalin restriction by decorating sequents with a term assignment. Hypotheses were assigned variables, and the conclusions were assigned terms. Then using these terms one can track the use of hypotheses throughout a derivation. They proposed a new implication right rule:

$$\frac{\Gamma, x : A \vdash t : B, \Delta \qquad x \notin \mathsf{FV}(\Delta)}{\Gamma \vdash \lambda x.t : A \multimap B, \Delta}\ \textsc{impR}$$

Intuitionism is enforced in this rule by requiring that the variable begin discharged, x, is potentially free in only one term annotating a conclusion. Unfortunately, this formalization did not enjoy cut-elimination either.

Bierman was able to give a counterexample to cut-elimination [1]. As Bierman explains the problem was with the left rule for par. The original rule was as follows:

$$\frac{\Gamma, x : A \vdash \Delta \qquad \Gamma', y : B \vdash \Delta'}{\Gamma, \Gamma', z : A \parr B \vdash \mathsf{let}\, z\, \mathsf{be}\, (x \parr -)\, \mathsf{in}\, \Delta \mid \mathsf{let}\, z\, \mathsf{be}\, (- \parr y)\, \mathsf{in}\, \Delta'}\ \textsc{parL}$$

In this rule the pattern variables x and y are bound in each term of L and L' respectively. Notice that the variable z becomes free in every term in L and L'. Bierman showed that this rule mixed with the restriction on implication right prevents the usual cut-elimination step that commutes cut with the left rule for par. The main idea behind the counterexample is that in the derivation before commuting the cut it is possible to discharge z using implication right, but after the cut is commuted past the left rule for par the variable z becomes free in more than one conclusion, and thus, can no longer be discharged.

In the conclusion of Bierman's note he gives an alternate left rule for par that he attributes to Bellin. This new left-rule is as follows:

$$\frac{\Gamma, x : A \vdash \Delta \quad \Gamma', y : B \vdash \Delta'}{\Gamma, \Gamma', z : A \parr B \vdash \text{let-pat}\, z\, (x \parr -)\, \Delta \mid \text{let-pat}\, z\, (- \parr y)\, \Delta'}\ \textsc{Parl}$$

In this rule let-pat $z\,(x \parr -)\,t$ and let-pat $z\,(- \parr y)\,t'$ only let-bind z in t or t' if $x \in FV(t)$ or $y \in FV(t')$. Otherwise the terms are left unaltered. Bellin conjectured that adopting this rule results in FILL regaining cut-elimination. However, no proof has been given.

Contributions. In this paper our main contribution is to confirm Bellin's conjecture by adopting his proposed rule and prove cut-elimination (Section 3). In addition, we show that this new formalization can be modeled by a new form of dialectica category called order-enriched dialectica category [3].

Related Work. [4]

2 Full Intuitionistic Linear Logic (FILL)

In this section we give a brief description of Full Intuitionistic Linear Logic (FILL) in the style found in [5]. However, we use a slightly different presentation that we feel provides a more elegant description of the logic. We first give the syntax of formulas, patterns, terms, and contexts. Following the syntax we define several meta-functions that will be used when defining the inference rules of the logic.

Definition 1. *The syntax for FILL is as follows:*

$$\begin{array}{rl}
\textit{(Formulas)} & A, B, C, D, E ::= I \mid \perp \mid A \multimap B \mid A \otimes B \mid A \parr B \\
\textit{(Patterns)} & p ::= * \mid - \mid x \mid p_1 \otimes p_2 \mid p_1 \parr p_2 \\
\textit{(Terms)} & t, e ::= x \mid * \mid \circ \mid t_1 \otimes t_2 \mid t_1 \parr t_2 \mid \lambda x.t \mid \mathsf{let}\ t\ \mathsf{be}\ p\ \mathsf{in}\ e \mid t_1\ t_2 \\
\textit{(Left Contexts)} & \Gamma ::= \cdot \mid x : A \mid \Gamma_1, \Gamma_2 \\
\textit{(Right Contexts)} & \Delta ::= \cdot \mid t : A \mid \Delta_1, \Delta_2
\end{array}$$

At this point we introduce some basic syntax and definitions to facilitate readability, and presentation of the inference rules. First, we will often write $\Delta_1 \mid \Delta_2$ as syntactic sugar for Δ_1, Δ_2. The former syntax should be read as "Δ_1 or Δ_2." This will help readability of the sequent we will introduce below. We denote the usual capture-avoiding substitution by $[t/x]t'$, and its extension to right contexts as $[t/x]\Delta$.

The previous extension will make conducting substitutions across a sequence of terms in an inference rule easier. Similarly, we find it convenient to be able to do this style of extension for the let-binding as well.

Definition 2. *We extend let-binding terms to right contexts as follows:*

$$
\begin{aligned}
&\mathsf{let}\, t\, \mathsf{be}\, p\, \mathsf{in}\, \cdot = \cdot \\
&\mathsf{let}\, t\, \mathsf{be}\, p\, \mathsf{in}\, (t' : A) = (\mathsf{let}\, t\, \mathsf{be}\, p\, \mathsf{in}\, t') : A \\
&\mathsf{let}\, t\, \mathsf{be}\, p\, \mathsf{in}\, (\Delta_1 \mid \Delta_2) = (\mathsf{let}\, t\, \mathsf{be}\, p\, \mathsf{in}\, \Delta_1) \mid (\mathsf{let}\, t\, \mathsf{be}\, p\, \mathsf{in}\, \Delta_2)
\end{aligned}
$$

We denote the usual function that computes the set of free variables in a term by $\mathsf{FV}(t)$, and its straightforward extension to right contexts as $\mathsf{FV}(\Delta)$. Finally, we arrive at the inference rules of FILL.

Definition 3. *The inference rules for derivability in FILL are as follows:*

$$
\frac{}{x : A \vdash x : A}\ \textsc{Ax} \qquad
\frac{\Gamma \vdash t : A \mid \Delta \quad y : A, \Gamma' \vdash \Delta'}{\Gamma, \Gamma' \vdash \Delta \mid [t/y]\Delta'}\ \textsc{Cut} \qquad
\frac{\Gamma \vdash \Delta}{\Gamma, x : I \vdash \mathsf{let}\, x\, \mathsf{be}\, *\, \mathsf{in}\, \Delta}\ \textsc{Il}
$$

$$
\frac{}{\cdot \vdash * : I}\ \textsc{Ir} \qquad
\frac{\Gamma, x : A, y : B \vdash \Delta}{\Gamma, z : A \otimes B \vdash \mathsf{let}\, z\, \mathsf{be}\, x \otimes y\, \mathsf{in}\, \Delta}\ \textsc{Tl}
$$

$$
\frac{\Gamma \vdash e : A \mid \Delta \quad \Gamma' \vdash f : B \mid \Delta'}{\Gamma, \Gamma' \vdash e \otimes f : A \otimes B \mid \Delta \mid \Delta'}\ \textsc{Tr} \qquad
\frac{}{x : \perp \vdash \cdot}\ \textsc{Pl} \qquad
\frac{\Gamma \vdash \Delta}{\Gamma \vdash \circ : \perp \mid \Delta}\ \textsc{Pr}
$$

$$
\frac{\Gamma, x : A \vdash \Delta \quad \Gamma', y : B \vdash \Delta'}{\Gamma, \Gamma', z : A \parr B \vdash \text{let-pat}\, z\, (x \parr -)\, \Delta \mid \text{let-pat}\, z\, (- \parr y)\, \Delta'}\ \textsc{Parl}
$$

$$
\frac{\Gamma \vdash \Delta \mid e : A \mid f : B \mid \Delta'}{\Gamma \vdash \Delta \mid e \parr f : A \parr B \mid \Delta'}\ \textsc{Parr} \qquad
\frac{\Gamma \vdash e : A \mid \Delta \quad \Gamma', x : B \vdash \Delta'}{\Gamma, y : A \multimap B, \Gamma' \vdash \Delta \mid [y\, e/x]\Delta'}\ \textsc{Impl}
$$

$$
\frac{\Gamma, x : A \vdash e : B \mid \Delta \quad x \notin \mathsf{FV}(\Delta)}{\Gamma \vdash \lambda x.e : A \multimap B \mid \Delta}\ \textsc{Impr} \qquad
\frac{\Gamma, x : A, y : B \vdash \Delta}{\Gamma, y : B, x : A \vdash \Delta}\ \textsc{Exl}
$$

$$
\frac{\Gamma \vdash \Delta_1 \mid t_1 : A \mid t_2 : B \mid \Delta_2}{\Gamma \vdash \Delta_1 \mid t_2 : B \mid t_1 : A \mid \Delta_2}\ \textsc{Exr}
$$

The Parl rule depends on a function let-pat $z\, p\, \Delta$. We define this function next.

Definition 4. *The function* let-pat $z\, p\, t$ *is defined as follows:*

$$
\begin{array}{ccc}
\text{let-pat}\, z\, (x \parr -)\, t = t & \text{let-pat}\, z\, (- \parr y)\, t = t & \text{let-pat}\, z\, p\, t = \mathsf{let}\, z\, \mathsf{be}\, p\, \mathsf{in}\, t \\
\textit{where } x \notin \mathsf{FV}(t) & \textit{where } y \notin \mathsf{FV}(t) &
\end{array}
$$

It is straightforward to extend the previous definition to right-contexts, and we denote this extension by let-pat $z\, p\, \Delta$.

The motivation behind this function is that it only binds the pattern variables in p in a term if and only if those pattern variables are free in the term. This over comes the counterexample given by Bierman in [1]. Throughout the sequel we will denote derivations of the previous rules by π.

3 Cut-elimination

The usual proof of cut-elimination for intuitionistic and classical linear logic should suffice for FILL. Thus, in this section we simply give the cut-elimination procedure for FILL following the development in [6]. However, there is one invariant that must be verified across each derivation transformation. The invariant is that if a derivation π is transformed into a derivation π', then the terms in the conclusion of the final rule applied in π must be equivalent to the terms in the conclusion of the final rule applied in π', but using what notion of equivalence?

Definition 5. *Equivalence on terms is defined as follows:*

$$\frac{y \notin \mathsf{FV}(t)}{t = [y/x]t}\ \textsc{Alpha} \qquad \frac{x \notin \mathsf{FV}(f)}{(\lambda x.f\ x) = f}\ \textsc{EtaFun} \qquad \frac{}{(\lambda x.e)\ e' = [e'/x]e}\ \textsc{BetaFun}$$

$$\frac{}{\mathsf{let}\ *\ \mathsf{be}\ *\ \mathsf{in}\ e = e}\ \textsc{Eta1I} \qquad \frac{}{\mathsf{let}\ u\ \mathsf{be}\ *\ \mathsf{in}\ [*/z]f = [u/z]f}\ \textsc{BetaI}$$

$$\frac{}{[\mathsf{let}\ u\ \mathsf{be}\ *\ \mathsf{in}\ e/y]f = \mathsf{let}\ u\ \mathsf{be}\ *\ \mathsf{in}\ [e/y]f}\ \textsc{NatI}$$

$$\frac{}{\mathsf{let}\ e \otimes t\ \mathsf{be}\ x \otimes y\ \mathsf{in}\ u = [e/x, t/y]u}\ \textsc{Beta1Ten}$$

$$\frac{}{\mathsf{let}\ u\ \mathsf{be}\ x \otimes y\ \mathsf{in}\ [x \otimes y/z]f = [u/z]f}\ \textsc{Beta2Ten}$$

$$\frac{}{[\mathsf{let}\ u\ \mathsf{be}\ x \otimes y\ \mathsf{in}\ g/w]f = \mathsf{let}\ u\ \mathsf{be}\ x \otimes y\ \mathsf{in}\ [g/w]f}\ \textsc{NatTen} \qquad \frac{}{u = \circ}\ \textsc{EtaParU}$$

$$\frac{}{(\mathsf{let}\ u\ \mathsf{be}\ x \parr -\ \mathsf{in}\ x) \parr (\mathsf{let}\ u\ \mathsf{be}\ - \parr y\ \mathsf{in}\ y) = u}\ \textsc{EtaPar}$$

$$\frac{}{\mathsf{let}\ u \parr t\ \mathsf{be}\ x \parr -\ \mathsf{in}\ e = [u/x]e}\ \textsc{Beta1Par} \qquad \frac{}{\mathsf{let}\ u \parr t\ \mathsf{be}\ - \parr y\ \mathsf{in}\ e = [t/y]e}\ \textsc{Beta2Par}$$

$$\frac{}{\mathsf{let}\ t\ \mathsf{be}\ x \parr -\ \mathsf{in}\ [u/x]f = [\mathsf{let}\ t\ \mathsf{be}\ x \parr -\ \mathsf{in}\ u/x]f}\ \textsc{Nat1Par}$$

$$\frac{}{\mathsf{let}\ t\ \mathsf{be}\ - \parr y\ \mathsf{in}\ [v/y]f = [\mathsf{let}\ t\ \mathsf{be}\ - \parr y\ \mathsf{in}\ v/y]f}\ \textsc{Nat2Par}$$

The cut elimination procedure requires the following two basic results:

Lemma 6 (Substitution Distribution)**.** *For any terms t, t_1, and t_2, $[t_1/x][t_2/y]t = [[t_1/x]t_2/y][t_2/x]t$.*

Proof. This proof holds by straightforward induction on the form of t. □

Lemma 7 (Let-pat Distribution)**.** *For any terms t, t_1, and t_2, and pattern p,* let-pat $t\ p\ [t_1/y]t_2 =$ [let-pat $t\ p\ t_1/y]t_2$.

Proof. This proof holds by case splitting over p, and then using the naturality equations for the respective pattern. □

Throughout the remainder of this section we present a particular step in the cut-elimination procedure, and then give a short proof that equality of terms are preserved across the particular transformation on derivations. Many of the transformations are trivial, and follow directly from the traditional proof. Thus, we only present here the most interesting cases. The full proof can be found in the companion report [?].

3.1 Principle formula vs. principle formula

3.1.1 Par

The proof

$$\dfrac{\textsc{Parr}\ \dfrac{\dfrac{\overset{\pi_1}{\vdots}}{\Gamma_1 \vdash \Delta_1 \mid t_1 : A \mid t_2 : B \mid \Delta_2}}{\Gamma_1 \vdash \Delta_1 \mid t_1 \parr t_2 : A \parr B \mid \Delta_2} \quad \dfrac{\dfrac{\overset{\pi_2}{\vdots}}{\Gamma_2, x : A \vdash \Delta_3} \qquad \dfrac{\overset{\pi_3}{\vdots}}{\Gamma_3, y : B \vdash \Delta_4}}{\Gamma_2, \Gamma_3, z : A \parr B \vdash \text{let-pat}\, z\, (x \parr -)\, \Delta_3 \mid \text{let-pat}\, z\, (- \parr y)\, \Delta_4}\ \textsc{ParL}}{\Gamma_2, \Gamma_3, \Gamma_1 \vdash \Delta_1 \mid \Delta_2 \mid [t_1 \parr t_2/z](\text{let-pat}\, z\, (x \parr -)\, \Delta_3) \mid [t_1 \parr t_2/z](\text{let-pat}\, z\, (- \parr y)\, \Delta_4)}\ \textsc{Cut}$$

is transformed into the proof

$$\dfrac{\dfrac{\dfrac{\dfrac{\overset{\pi_1}{\vdots}}{\Gamma_1 \vdash \Delta_1 \mid t_1 : A \mid t_2 : B \mid \Delta_2} \qquad \dfrac{\overset{\pi_3}{\vdots}}{\Gamma_3, y : B \vdash \Delta_4}}{\Gamma_3, \Gamma_1 \vdash \Delta_1 \mid t_1 : A \mid \Delta_2 \mid [t_2/y]\Delta_4}\ \textsc{Cut} \qquad \dfrac{\overset{\pi_2}{\vdots}}{\Gamma_2, x : A \vdash \Delta_3}}{\Gamma_2, \Gamma_3, \Gamma_1 \vdash \Delta_1 \mid \Delta_2 \mid [t_2/y]\Delta_4 \mid [t_1/x]\Delta_3}\ \textsc{Cut}}{\Gamma_2, \Gamma_3, \Gamma_1 \vdash \Delta_1 \mid \Delta_2 \mid [t_1/x]\Delta_3 \mid [t_2/y]\Delta_4}\ \textsc{Series of Exchanges}$$

Without loss of generality consider the case when $\Delta_3 = t_3 : C_1 \mid \Delta_3'$ and $\Delta_4 = t_4 : C_2 \mid \Delta_4'$. First, $[t_1 \parr t_2/z](\text{let-pat}\, z\, (x \parr -)\, t_3) = \text{let-pat}\, (t_1 \parr t_2)\, (x \parr -)\, t_3$, and by $\textsc{Eq\_Beta1Par}$ we know $\text{let-pat}\, (t_1 \parr t_2)\, (x \parr -)\, t_3 = [t_1/x]t_3$ if $x \in \mathsf{FV}(t_3)$ or $\text{let-pat}\, (t_1 \parr t_2)\, (x \parr -)\, t_3 = t_3$ otherwise. In the latter case we can see that $t_3 = [t_1/x]t_3$, thus, in both cases $\text{let-pat}\, (t_1 \parr t_2)\, (x \parr -)\, t_3 = [t_1/x]t_3$. This argument can be repeated for any terms in Δ_3', and hence $[t_1 \parr t_2/z](\text{let-pat}\, z\, (x \parr -)\, \Delta_3) = \text{let-pat}\, (t_1 \parr t_2)\, (x \parr -)\, \Delta_3 = [t_1/x]\Delta_3$. We can apply a similar argument for $[t_1 \parr t_2/z](\text{let-pat}\, z\, (- \parr y)\, t_4)$ and $[t_1 \parr t_2/z](\text{let-pat}\, z\, (- \parr y)\, \Delta_4)$.

Note that we could have first cut on A, and then on B in the second derivation, but we would have arrived at the same result just with potentially more exchanges on the right.

3.1.2 Implication

The proof

$$\dfrac{\dfrac{\dfrac{\overset{\pi_1}{\vdots}}{\Gamma, x : A \vdash t : B \mid \Delta} \qquad x \notin \mathsf{FV}(\Delta)}{\Gamma \vdash \lambda x.t : A \multimap B \mid \Delta}\ \textsc{ImpR} \qquad \dfrac{\dfrac{\overset{\pi_2}{\vdots}}{\Gamma_1 \vdash t_1 : A \mid \Delta_1} \qquad \dfrac{\overset{\pi_3}{\vdots}}{\Gamma_2, y : B \vdash \Delta_2}}{\Gamma_1, z : A \multimap B, \Gamma_2 \vdash \Delta_1 \mid [z\, t_1/y]\Delta_2}\ \textsc{ImpL}}{\Gamma_1, \Gamma, \Gamma_2 \vdash \Delta \mid [\lambda x.t/z]\Delta_1 \mid [\lambda x.t/z][z\, t_1/y]\Delta_2}\ \textsc{Cut}$$

transforms into the proof

$$
\dfrac{\dfrac{\dfrac{\begin{matrix}\pi_2\\ \vdots\end{matrix}}{\Gamma_1 \vdash t_1 : A \mid \Delta_1} \quad \dfrac{\begin{matrix}\pi_1\\ \vdots\end{matrix}}{\Gamma, x : A \vdash t : B \mid \Delta} \quad x \notin \mathsf{FV}(\Delta)}{\Gamma, \Gamma_1 \vdash \Delta_1 \mid [t_1/x]t : B \mid [t_1/x]\Delta}\;\textsc{Cut} \qquad \dfrac{\begin{matrix}\pi_3\\ \vdots\end{matrix}}{\Gamma_2, y : B \vdash \Delta_2}}{\Gamma_2, \Gamma, \Gamma_1 \vdash \Delta_1 \mid [t_1/x]\Delta \mid [[t_1/x]t/y]\Delta_2}\;\textsc{Cut}}{\Gamma_1, \Gamma, \Gamma_2 \vdash [t_1/x]\Delta \mid \Delta_1 \mid [[t_1/x]t/y]\Delta_2}\;\textsc{Series of Exchanges}
$$

Without loss of generality consider the case when $\Delta_2 = t_2 : C \mid \Delta'_2$. First, by hypothesis we know $x \notin \mathsf{FV}(\Delta)$, and so we know $\Delta = [t_1/x]\Delta$. We can see that $[\lambda x.t/z][z\ t_1/y]t_2 = [(\lambda x.t)\ t_1/y]t_2 = [[t_1/x]t/y]t_2$ by using the congruence rules of equality and the rule Eq_BetaFun. This argument can be repeated for any term in $[\lambda x.t/z][z\ t_1/y]\Delta'_2$, and so $[\lambda x.t/z][z\ t_1/y]\Delta_2 = [[t_1/x]t/y]\Delta_2$. Finally, by inspecting the previous derivations we can see that $z \notin \mathsf{FV}(\Delta_1)$, and thus, $\Delta_1 = [\lambda x.t/z]\Delta_1$.

References

[1] Gavin Bierman. A note on full intuitionistic linear logic. *Annals of Pure and Applied Logic*, 79(3):281 – 287, 1996.

[2] Valeria de Paiva. Dialectica categories. In J. Gray and A. Scedrov, editors, *Categories in Computer Science and Logic*, volume 92, pages 47–62. Amerian Mathemaitcal Society, 1989.

[3] Valeria de Paiva. Bounded dialectica interpretation: categorically. Under review, 2015.

[4] Valeria de Paiva and Luiz Carlos Pereira. A short note on intuitionistic propositional logic with multiple conclusions. *MANUSCRITO - Rev. Int. Fil.*, 28(2):317 – 329, jul-dez 2005.

[5] Martin Hyland and Valeria de Paiva. Full intuitionistic linear logic (extended abstract). *Annals of Pure and Applied Logic*, 64(3):273 – 291, 1993.

[6] Paul-Andre Mellies. *Categorical Semantics of Linear Logic.* 2009.

[7] Harold Schellinx. Some syntactical observations on linear logic. *Journal of Logic and Computation*, 1(4):537–559, 1991.