A Family of Adjoint Models as the Foundation of Process Tree Based Threat Analysis

Harley Eades III[1] and Jiaming Jiang[2]

1 **Computer Science, Augusta University, Augusta, Georgia, USA**
`heades@augusta.edu`
2 **Computer Science, North Carolina State University, Raleigh, North Carolina, USA**
`jjiang13@ncsu.edu`

Abstract

TODO

1998 ACM Subject Classification TODO

Keywords and phrases TODO

Digital Object Identifier 10.4230/LIPIcs...

1 Introduction

TODO [1]

2 Categorical Models

We develop a categorical framework in which many different intuitionistic substructural logics may be modeled. The locus of this framework is an adjunction. We initially take a monoidal category, $\mathcal{L}$, as a base, and then extend it with one or more structural morphisms – a morphism corresponding to a structural rule in logic – to obtain a second category $\hat{\mathcal{L}}$. Then we form a monoidal adjunction $\hat{\mathcal{L}} : F \dashv G : \mathcal{L}$ just as Benton [1] did for intuitionistic linear logic. Depending on which structural morphisms we add to $\hat{\mathcal{L}}$ we will obtain different models. In particular, each model will come endowed with a comonad on $\mathcal{L}$ which equips $\mathcal{L}$ with the ability to track the corresponding structural rule(s).

We will show that by adding the morphisms for either weakening, contraction, or exchange, to $\mathcal{L}$ will yield an adjoint model of non-commutative relevance logic/linear logic, non-commutative contraction logic/linear logic, and commutative/non-commutative linear logic. The latter model will come with a monoidal comonad $e : \mathcal{L} \longrightarrow \mathcal{L}$ such that there is a symmetry $\mathsf{ex}_{A,B} : eA \triangleright eB \longrightarrow eB \triangleright eA$, where $- \triangleright -$ denotes a non-commutative tensor product. In fact, this is the first adjoint model of the Lambek calculus with the exchange comonad.

At this point we will have adjoint models for each individual structural rule. What if we want more than one structural rule? There are a few different choices that one can choose from depending on the scenario. First, if $\hat{\mathcal{L}}$ contains more than one structural morphism, then $\mathcal{L}$ will have a single comonad that adds all of those structural morphisms to $\mathcal{L}$. For example, if $\hat{\mathcal{L}}$ contains weakening, contraction, and exchange, then $\hat{\mathcal{L}}$ is cartesian closed and $\mathcal{L}$ will have the usual $! : \mathcal{L} \longrightarrow \mathcal{L}$ comonad. The second scenario is when $\mathcal{L}$ also contains some structural morphisms. For example, if $\hat{\mathcal{L}}$ contains exchange and weakening and $\mathcal{L}$ contains exchange, then $\mathcal{L}$ will have a comonad, $r : \mathcal{L} \longrightarrow \mathcal{L}$, which combines linear logic with relevance logic. Thus, how we instantiate the two categories in the adjunction influences which logic one may model.

What if we want multiple comonads tracking different logics? In this scenario the different comonads would allow us to mix the different logics in interesting ways. Suppose $\mathcal{L}$ has no structural

Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

morphisms and $\mathcal{E}$ is $\mathcal{L}$ with exchange and $\mathcal{EW}$ is $\mathcal{E}$ with weakening. Then we can form two adjunctions $\mathcal{E} : F \dashv G : \mathcal{L}$ and $\mathcal{EW} : H \dashv J : \mathcal{L}$, but the categories $\mathcal{E}$ and $\mathcal{EW}$ have a structural morphism in common. So instead, we form the adjunction $\mathcal{EW} : H \dashv J : \mathcal{E} : F \dashv G : \mathcal{L}$. Thus, $\mathcal{L}$ has the exchange comonad $e = FG : \mathcal{L} \longrightarrow \mathcal{L}$ as well as the relevance logic comonad $r = FHJG : \mathcal{L} \longrightarrow \mathcal{L}$. Additionally, there is a comonad $w = JH : \mathcal{E} \longrightarrow \mathcal{E}$ adding weakening to $\mathcal{E}$. This idea is based on the amazing work of Mellies [4]. Throughout the remainder of this section we make these ideas precise.

2.1 Lambek Categories

The bases of all of our models will be what we call Lambek categories. These are named after Joachim Lambek to pay homage to his work on the Lambek calculus which can be seen as non-commutative intuitionistic linear logic [3]. Thus, each of our models have a very basic foundation.

Lambek categories are based on (non-symmetric) monoidal categories.

▶ **Definition 1.** A **monoidal category**, $(\mathcal{L}, \triangleright, I, \lambda, \rho)$, is a category, $\mathcal{L}$, equipped with a bifunctor, $\triangleright : \mathcal{L} \times \mathcal{L} \longrightarrow \mathcal{L}$, called the tensor product, a distinguished object I of $\mathcal{L}$ called the unit, and three natural isomorphisms $\lambda_A : I \triangleright A \longrightarrow A$, $\rho_A : A \triangleright I \longrightarrow A$, and $\alpha_{A,B,C} : A \triangleright (B \triangleright C) \longrightarrow (A \triangleright B) \triangleright C$ called the left and right unitors and the associator respectively. Finally, these are subject to the following coherence diagrams:

$$\begin{tikzcd}
((A \triangleright B) \triangleright C) \triangleright D \arrow[r, "\alpha_{A,B,C} \triangleright \mathsf{id}_D"] \arrow[d, "\alpha_{A \triangleright B,C,D}"'] & (A \triangleright (B \triangleright C)) \triangleright D \arrow[r, "\alpha_{A,B \triangleright C,D}"] & A \triangleright ((B \triangleright C) \triangleright D) \arrow[d, "\mathsf{id}_A \triangleright \alpha_{B,C,D}"] \\
(A \triangleright B) \triangleright (C \triangleright D) \arrow[rr, "\alpha_{A,B,C \triangleright D}"] & & A \triangleright (B \triangleright (C \triangleright D))
\end{tikzcd}$$

$$\begin{tikzcd}
(A \triangleright I) \triangleright B \arrow[rr, "\alpha_{A,I,B}"] \arrow[dr, "\rho_A \triangleright id_B"'] & & A \triangleright (I \triangleright B) \arrow[dl, "id_A \triangleright \lambda_B"] \\
& A \triangleright B &
\end{tikzcd}$$

A Lambek category adds closure to monoidal categories.

▶ **Definition 2.** A **Lambek category** is a monoidal category $(\mathcal{L}, \triangleright, I, \lambda, \rho, \alpha)$ equipped with two bifunctors $\rightharpoonup : \mathcal{L}^{\mathsf{op}} \times \mathcal{L} \longrightarrow \mathcal{L}$ and $\leftharpoonup : \mathcal{L} \times \mathcal{L}^{\mathsf{op}} \longrightarrow \mathcal{L}$ that are both right adjoint to the tensor product. That is, the following natural bijections hold:

$$\mathsf{Hom}_{\mathcal{L}}(X \triangleright A, B) \cong \mathsf{Hom}_{\mathcal{L}}(X, A \rightharpoonup B) \qquad \mathsf{Hom}_{\mathcal{L}}(A \triangleright X, B) \cong \mathsf{Hom}_{\mathcal{L}}(X, B \leftharpoonup A)$$

An alternative name for Lambek categories is biclosed monoidal categories.

If we add add a symmetry to a Lambek category then we will obtain a symmetric monoidal closed category. The following two definitions and lemma capture this result.

▶ **Definition 3.** A monoidal category $(\mathcal{L}, \triangleright, I, \lambda, \rho, \alpha)$ is **symmetric** if there is a natural transformation $\mathsf{ex}_{A,B} : A \triangleright B \longrightarrow B \triangleright A$ such that $\mathsf{ex}_{B,A} \circ \mathsf{ex}_{A,B} = \mathsf{id}_{A \triangleright B}$ and the following commute:

$$\begin{tikzcd}
(A \triangleright B) \triangleright C \arrow[r, "\alpha_{A,B,C}"] \arrow[d, "\mathsf{ex}_{A,B} \triangleright \mathsf{id}_C"'] & A \triangleright (B \triangleright C) \arrow[r, "\mathsf{ex}_{A,B \triangleright C}"] & (B \triangleright C) \triangleright A \arrow[d, "\alpha_{B,A,C}"] \\
(B \triangleright A) \triangleright C \arrow[r, "\alpha_{B,A,C}"] & B \triangleright (A \triangleright C) \arrow[r, "\mathsf{id}_B \triangleright \mathsf{ex}_{A,C}"] & B \triangleright (C \triangleright A)
\end{tikzcd}
\qquad
\begin{tikzcd}
I \triangleright A \arrow[rr, "\mathsf{ex}_{I,A}"] \arrow[dr, "\lambda_A"'] & & A \triangleright I \arrow[dl, "\rho_A"] \\
& A &
\end{tikzcd}$$

Throughout this paper when $- \triangleright -$ is symmetric we denote it by $- \otimes -$.

We call a symmetric Lambek category a Lambek category with exchange, because the symmetric models the exchange rule.

▶ **Definition 4.** A symmetric monoidal category $(\mathcal{L}, \otimes, I, \lambda, \rho, \alpha, \beta)$ is **closed** if it comes equipped with a bifunctor $\multimap : \mathcal{L}^{\mathsf{op}} \times \mathcal{L} \longrightarrow \mathcal{L}$ that is right adjoint to the tensor product. That is, the following natural bijection $\mathsf{Hom}_{\mathcal{L}}(X \otimes A, B) \cong \mathsf{Hom}_{\mathcal{L}}(X, A \multimap B)$ holds.

▶ **Lemma 5.** *Let A and B be two objects in a Lambek category with exchange. Then $(A \rightharpoonup B) \cong (B \leftharpoonup A)$.*

Proof. First, notice that for any object C we have

$$
\begin{array}{rcll}
Hom[C, A \rightharpoonup B] & \cong & Hom[C \otimes A, B] & \mathcal{L} \text{ is a Lambek category} \\
& \cong & Hom[A \otimes C, B] & \text{By the symmetry } \mathsf{ex}_{C,A} \\
& \cong & Hom[C, B \leftharpoonup A] & \mathcal{L} \text{ is a Lambek category}
\end{array}
$$

Thus, $A \rightharpoonup B \cong B \leftharpoonup A$ by the Yoneda lemma. ◀

▶ **Corollary 6.** *A Lambek category with exchange is symmetric monoidal closed.*

We will also be discussing two other structural rules: weakening and contraction. These are defined as follows.

▶ **Definition 7.** A **Lambek category with weakening**, $(\mathcal{L}, \otimes, I, \lambda, \rho, \alpha, \mathsf{weak})$, is a Lambek category $(\mathcal{L}, \otimes, I, \lambda, \rho, \alpha)$ equipped with a natural transformation $\mathsf{weak}_A : A \longrightarrow I$.

▶ **Definition 8.** A **Lambek category with contraction**, $(\mathcal{L}, \otimes, I, \lambda, \rho, \alpha, \mathsf{contraL}, \mathsf{contraR})$, is a Lambek category $(\mathcal{L}, \otimes, I, \lambda, \rho, \alpha)$ equipped with natural transformations:

$$
\mathsf{contraL}_{A,B} : (A \otimes B) \longrightarrow (A \otimes B) \otimes A \qquad \mathsf{contraR}_{A,B} : (B \otimes A) \longrightarrow A \otimes (B \otimes A)
$$

Furthermore, the following diagrams must commute:

$$
\begin{tikzcd}
A \otimes I \arrow[d, "\mathsf{contraL}_{A,I}"'] & A \arrow[l, "\rho_A^{-1}"'] \arrow[r, "\lambda_A^{-1}"] & I \otimes A \arrow[d, "\mathsf{contraR}_{A,I}"] \\
(A \otimes I) \otimes A \arrow[rr, "\alpha_{A,I,A}"'] & & A \otimes (I \otimes A)
\end{tikzcd}
$$

$$
\begin{tikzcd}
A \otimes A \arrow[r, "id_A \otimes \rho_A^{-1}"] \arrow[dd, "\lambda_A^{-1} \otimes id_A"'] & A \otimes (A \otimes I) \arrow[r, "id_A \otimes \mathsf{contraL}_{A,I}"] & A \otimes ((A \otimes I) \otimes A) \arrow[d, "id_A \otimes (\rho_A \otimes id_A)"] \\
& & A \otimes (A \otimes A) \\
(I \otimes A) \otimes A \arrow[r, "\mathsf{contraR}_{A,I} \otimes id_A"'] & (A \otimes (I \otimes A)) \otimes A \arrow[r, "(id_A \otimes \lambda_A) \otimes id_A"'] & (A \otimes A) \otimes A \arrow[u, "\alpha_{A,A,A}"']
\end{tikzcd}
$$

We call the morphisms:

$\mathsf{ex}_{A,B} : A \otimes B \longrightarrow B \otimes A$
$\mathsf{weak}_A : A \longrightarrow I$
$\mathsf{contraL}_{A,B} : (A \otimes B) \longrightarrow (A \otimes B) \otimes A$
$\mathsf{contraR}_{A,B} : (B \otimes A) \longrightarrow A \otimes (B \otimes A)$

structural morphisms, because they all model the various structural rules in intuitionistic logic.

2.2 Structural Adjoint Models

Now we turn to making our model precise.

▶ **Definition 9.** Suppose $\mathcal{L}_0, \ldots, \mathcal{L}_n$ is a family of Lambek categories with zero or more structural morphisms where $\mathcal{L}_i$ is a full subcategory of $\mathcal{L}_{i+1}$. Then a **structural adjoint model**, $\overrightarrow{\mathcal{L}_n : F_n \dashv G_n : \mathcal{L}_{n-1}}$, is a composition of monoidal adjunctions:

$$\mathcal{L}_n : F_n \dashv G_n : \mathcal{L}_{n-1} : F_{n-1} \dashv G_{n-1} : \mathcal{L}_{n-2} : \cdots : \mathcal{L}_1 : F \dashv G : \mathcal{L}_0.$$

This definition is an extension – or perhaps a simplification due to the isolation of exchange – of the models discussed by Melliés [4].

Our structural adjoint model subsumes Benton's [1] linear/non-linear model (LNL). Simply take the sequence of Lambek categories to be $\mathcal{L}_0$, a Lambek category with exchange, and $\mathcal{L}_1$, a Lambek category with weakening, contraction, and exchange, and thus, $\mathcal{L}_1$ is cartesian closed. However, our model is a lot more flexible and expressive.

▶ **Lemma 10** (Comonads in a Structural Adjoint Model)**.** *Suppose $\overrightarrow{\mathcal{L}_n : F_n \dashv G_n : \mathcal{L}_{n-1}}$ is a structural adjoint model. Then there are the following comonads:*

- $(\mathcal{L}_0, F_0G_0, \varepsilon^0, \delta^0), \ldots, (\mathcal{L}_{n-1}, F_nG_n, \varepsilon^n, \delta^n)$
- $(\mathcal{L}_0, F_0F_1G_1G_0, \varepsilon^0, \delta^0), \ldots, (\mathcal{L}_{n-1}, F_{n-1}F_nG_nG_{n-1}, \varepsilon^n, \delta^n)$
- $(\mathcal{L}_0, F_0F_1F_2G_2G_1G_0, \varepsilon^0, \delta^0), \ldots, (\mathcal{L}_{n-1}, F_{n-2}F_{n-1}F_nG_nG_{n-1}G_{n-2}, \varepsilon^n, \delta^n)$

  $\vdots$
- $(\mathcal{L}_0, F_0 \cdots F_nG_n \cdots G_0, \varepsilon^0, \delta^0)$

Proof. This proof easily follows from the well-known fact that adjoints induce comonads – as well as monads – and composition of adjoints. ◀

The previous lemma shows that a Lambek category $\mathcal{L}_i$ in the sequence is endowed with all of the structure found in each of the categories above it, but this structure is explicitly tracked using the various comonads. That is, the Eilenberg-Moore category of each of the comonads mentioned in the previous lemma has the corresponding structural rule as morphisms.

▶ **Lemma 11.** *Suppose $\mathcal{L}_l : F \dashv G : \mathcal{L}_r$ is a monoidal adjunction in the structural adjoint model $\overrightarrow{\mathcal{L}_n : F_n \dashv G_n : \mathcal{L}_{n-1}}$. Then the Eilenberg-Moore category, $\mathcal{L}_r^E$, contains all of the structural morphisms from both $\mathcal{L}_r$ and $\mathcal{L}_l$.*

Proof. This result holds similarly to Benton's [1] proof that the Eilenberg-Moore category for the of-course comoand is cartesian closed. So we omit the details. ◀

2.3 Example Structural Adjoint Models

We can give a number of example adjoint structures that are of interest to the research community.

Lambek Calculus with Exchange. The first is a model that reveals how to combine the Lambek Calculus with the exchange comonad and Girard's of-course comonad. This model is of interest to the linguistics community [**?**], because they often only want exchange in very controlled instances. Valeria de Paiva [**?**] was the first to show that this is possible using Dialectica Categories and Reedy's [**?**] model. However, she uses a comonad with the natural transformation $\mathsf{ex}_{A,B} : A \triangleright eB \longrightarrow eB \triangleright A$, but we feel this goes against the standard view of algebraic binary operations. In addition, while Dialectica categories are extremely useful, but rather complex, we are interested in simpler models. Thus, we prefer an adjoint model with a comonad which has the natural transformation $\mathsf{ex}_{A,B} : eA \triangleright eB \longrightarrow eB \triangleright eA$.

As we have said in the introduction there are many security applications where one must have both a commutative and a non-commutative tensor product within the same logic. For example, when reasoning about process trees in threat analysis.

▶ **Definition 12.** Suppose $\mathcal{L}_{ewc}$ is a Lambek category with exchange, weakening, and contraction, $\mathcal{L}_e$ is a Lambek category with exchange, and $\mathcal{L}$ is a Lambek category. Then a **LC adjoint model** is the structural adjoint model $\mathcal{L}_{ewc} : H \dashv J : \mathcal{L}_e : F \dashv G : \mathcal{L}$.

We must now show that $\mathcal{L}$ in a LC adjoint model has two comonads $e : \mathcal{L} \longrightarrow \mathcal{L}$ adding exchange to $\mathcal{L}$, and $! : \mathcal{L} \longrightarrow \mathcal{L}$ – Girard's of-course modality – adding weakening, contraction, and exchange. We first have the following corollary to Lemma 10.

▶ **Corollary 13.** *Suppose $\mathcal{L}_{ewc} : H \dashv J : \mathcal{L}_e : F \dashv G : \mathcal{L}$ is a LC adjoint model. Then there are comonads $(e : \mathcal{L} \longrightarrow \mathcal{L}, \varepsilon^e, \delta^e)$, $(! : \mathcal{L} \longrightarrow \mathcal{L}, \varepsilon^!, \delta^!)$, and $(!_e : \mathcal{L}_e \longrightarrow \mathcal{L}_e, \varepsilon^{!_e}, \delta^{!_e})$.*

Proof. We only show how each of the tuples are defined:

- $eA = FGA$, $\varepsilon^e_A : eA \longrightarrow A$ is the counit of the adjunction, and $\delta^e_A = F\eta^e_{GA} : eA \longrightarrow eeA$, where $\eta^e_A : A \longrightarrow GFA$ is the unit of the adjunction.
- $!_eA = HJA$, $\varepsilon^{!_e}_A : !_eA \longrightarrow A$ is the counit of the adjunction, and $\delta^{!_e}_A = H\eta^{!_e}_{JA} : !_eA \longrightarrow !_e!_eA$, where $\eta^{!_e}_A : A \longrightarrow JHA$ is the unit of the adjunction.
- $!A = FHJGA$, $\varepsilon^!_A : !A \longrightarrow A$ is the counit of the adjunction, and $\delta^!_A = FH\eta^!_{JGA} : !A \longrightarrow !!A$, where $\eta^!_A : A \longrightarrow JGFHA$ is the unit of the adjunction.

◀

As a corollary to Lemma 11 we show that the Eilenberg-Moore categories contain the required structure.

▶ **Corollary 14.** *Suppose $\mathcal{L}_{ewc} : H \dashv J : \mathcal{L}_e : F \dashv G : \mathcal{L}$ is a LC adjoint model. Then the Eilenberg-Moore Categories associated with the comonads $(e : \mathcal{L} \longrightarrow \mathcal{L}, \varepsilon^e, \delta^e)$, $(! : \mathcal{L} \longrightarrow \mathcal{L}, \varepsilon^!, \delta^!)$, and $(!_e : \mathcal{L}_e \longrightarrow \mathcal{L}_e, \varepsilon^{!_e}, \delta^{!_e})$ have the structure: $\mathcal{L}^E_e$ is symmetric monoidal, and $\mathcal{L}^E_!$ and $\mathcal{L}^E_{!_e}$ are cartesian closed.*

Proof. The proof that $\mathcal{L}^E_{!_e}$ is cartesian closed follows from Benton [1], because $\mathcal{L}_e$ is symmetric monoidal, and $\mathcal{L}_{ewc}$ is cartesian closed, and hence $\mathcal{L}_{ewc} : H \dashv J : \mathcal{L}_e$ is a LNL model. So we only give proofs of the other two categories.

TODO: Jiaming ◀

The previous result shows that $\mathcal{L}$ has the following structural morphisms:

$$\mathsf{ex}_{A,B} : eA \triangleright eB \longrightarrow eB \triangleright eA \qquad \mathsf{weak}_A : !A \longrightarrow I \qquad \mathsf{contra}_A : !A \triangleright !A \longrightarrow !A$$

In addition, the category $\mathcal{L}_e$ has the following structural morphisms:

$$\mathsf{ex}_{A,B} : A \otimes B \longrightarrow B \otimes A \qquad \mathsf{weak}_A : !_eA \longrightarrow I \qquad \mathsf{contra}_A : !_eA \otimes !_eA \longrightarrow !_eA$$

However, notice that exchange is a first class citizen, and hence, is not tracked by a comonad.

Affine Logic. Affine logic has applications in verification of security protocols [2]. Structural adjoint models provide a means to combine intuitionistic linear logic with affine logic.

▶ **Definition 15.** Suppose $\mathcal{L}_{ew}$ is a Lambek category with exchange and weakening, and $\mathcal{L}_e$ is a Lambek category with exchange. Then an **affine adjoint model** is the structural adjoint model $\mathcal{L}_{ew} : F \dashv G : \mathcal{L}_e$.

Just as above this model comes with a comonad $(w : \mathcal{L}_e \longrightarrow \mathcal{L}_e, \varepsilon^w, \delta^w)$ defined in the same way as $!_e A$ from above. This then equips $\mathcal{L}_e$ with the natural transformation $\mathsf{weak}_A : wA \longrightarrow I$. Finally, the Eilenberg-Moore category $\mathcal{L}_w^E$ models intuitionistic affine logic.

In this example we made both categories symmetric, but this is not strictly necessary. One could also model non-commutative affine logic as well. In fact, commutativity – as well as all of the other structural rules – is optional in every logic we discuss in this paper.

Strict Logic. Similarly to affine logic we can model strict logic – sometimes referred to as contraction logic – as well.

▶ **Definition 16.** Suppose $\mathcal{L}_{ec}$ is a Lambek category with exchange and contraction, and $\mathcal{L}_e$ is a Lambek category with exchange. Then a **strict adjoint model** is the structural adjoint model $\mathcal{L}_{ec} : F \dashv G : \mathcal{L}_e$.

Just as above this model comes with a comonad $(c : \mathcal{L}_e \longrightarrow \mathcal{L}_e, \varepsilon^c, \delta^c)$ defined in the same way as $!_e A$ and wA from above. This then equips $\mathcal{L}_e$ with the natural transformation $\mathsf{contra}_A : cA \otimes cA \longrightarrow cA$. Finally, the Eilenberg-Moore category $\mathcal{L}_c^E$ models intuitionistic strict logic.

Combining Logics.

3 Related Work

TODO

4 Conclusion

TODO

References

1 P. N. Benton. A mixed linear and non-linear logic: Proofs, terms and models (preliminary report). Technical Report UCAM-CL-TR-352, University of Cambridge Computer Laboratory, 1994. Accessible online at `http://research.microsoft.com/en-us/um/people/nick/mixed3.ps`.

2 Michele Bugliesi, Stefano Calzavara, Fabienne Eigner, and Matteo Maffei. Affine refinement types for secure distributed programming. *ACM Trans. Program. Lang. Syst.*, 37(4):11:1–11:66, August 2015. URL: `http://doi.acm.org/10.1145/2743018`, `doi:10.1145/2743018`.

3 Joachim Lambek. The mathematics of sentence structure. *Journal of Symbolic Logic*, 33(4):627–628, 1968.

4 Paul-André Melliès. Comparing hierarchies of types in models of linear logic. *Information and Computation*, 189(2):202 – 234, 2004. `doi:http://dx.doi.org/10.1016/j.ic.2003.10.003`.

A Appendix