

Rapport Projet Outils pour le raisonnement propositionnel : Structures de données arborescentes

L2 Informatique - Université de Rouen

Année 2023 - 2024

Bossut Romaric et Hery Yohan

Table des matières

1	Introduction	2
1.1	Présentation du sujet	2
1.2	Problématiques	2
2	Présentation de notre projet	2
2.1	Fonctions supplémentaires	2
2.2	Méthodologie	3
3	Exemples d'exécution de Tseitin	3
3.1	Contenu des fichiers	3
4	Conclusion	4

1 Introduction

1.1 Présentation du sujet

Le sujet nous demande de développer la transformation de Tseitin due au mathématicien Grigori Tseitin, en OCaml. C'est une méthode permettant de modifier une formule afin d'éviter une explosion combinatoire potentielle lors de la mise en forme clausale conjonctive. Nous devons aussi développer des méthodes permettant de tester empiriquement nos résultats à l'aide de formules construites (pseudo-)aléatoirement.

Pour cela, nous avons eu recours à différents modules rédigés durant le semestre qui ont pu être réutilisés :

- Adaptation de chacun des modules due au rajout de la formule Equiv qui représente l'équivalence et prend deux formules en paramètre.
- Rajout des fonctions manquantes.
- Création de fonctions annexes

1.2 Problématiques

Nous devons adapter les différentes fonctions des modules suivants :

- FCC
- Formule
- Quine
- DPLL

Ajouter la formule Equiv ne change pas énormément de chose dans l'application de la méthode car dans certaines méthode le Equiv ne doit pas ou est déjà traiter de la même manière que Et, Ou, Imp.

2 Présentation de notre projet

2.1 Fonctions supplémentaires

```
1  Tseitin :
2      old_or_new;
3      atop;
4  CorrecTest :
5      contain_equiv;
6      true_list;
7      create_inter;
8  RandomFormule :
9      random_atome;
```

2.2 Méthodologie

Pour développer les différentes fonctions demandées dans le projet, nous avons utilisé les différentes indications données dans le sujet pour certaines fonctions et pour d'autres, on a suivis une logique (ex : Tseitin) qui à été expliqué en partie dans le sujet mais aussi utilisé des exemples pour pouvoir faire en sorte qu'elles répondent à nos attentes. Certaines fonctions sont tous simplement des réadaptations de fonctions travaillées lors de TP et TD.

3 Exemples d'exécution de Tseitin

```
Type #utop_help for help about using utop.
-( 22:28:47 )-< command 0 >
utop # string_of_formule (tseitin (Et(Ou(Atome "a", Atome "b"), Non(Atome "a"))));
- : string =
"(new_0 ∧ ((new_0 ↔ (new_1 ∧ new_2)) ∧ ((new_1 ↔ (old_a ∨ old_b)) ∧ (new_2 ↔ ¬old_a))))"
-( 22:28:47 )-< command 1 >
utop #
```

3.1 Contenu des fichiers

Formule

Création du type formule, du type interpretation et de leurs composants. Leur fonctions pour évaluer les interprétations et transformer des formules dans le type string pour pouvoir les afficher ultérieurement.

Benchmark

Calcule du temps d'exécution des différentes fonctions

FCC

Différentes fonctions traitant des formules afin d'obtenir leur forme ensembliste (forme clausale conjonctive : FCC)

DPLL

Fonction de simplification des FCC et tests de satisfaisabilité par propagation unitaire ou classiquement DPLL.

Quine

Fonction mettant en forme l'algorithme de Quine (choix d'un atome, substitution, test de satisfaisabilité) où l'on doit d'abord simplifier une formule pour pouvoir l'exécuter.

RandomFormule

Module de création d'une formule de manière aléatoire via une sélection aléatoire d'atome dans un alphabet.

CorrecTest

Fais un test d'équisatisfaisabilité, appliqué sur les algorithmes de Quine ou DPLL, via différentes fonctions sur une formule et sa transformée de Tseitin.

4 Conclusion

Pour conclure, nous avons eu, au cours de ce projet, qu'un seul problème pour la fonction extension que nous avons eu du mal à visualiser. Sinon, dans l'ensemble, tout c'est enchainé assez rapidement. Nous avons repris nos travaux de TP pour certains fichiers (Quine, DPLL, Formule, FCC) qui nous a permis un gain de temps. Ensuite, pour ce qu'il reste, nous avons fait preuve de réflexion à l'aide d'exemple sur feuille nous permettant de mener à bien ce projet et d'écrire étape par étape les différentes fonctions.