

**GERAD** Groupe d'études et de recherche en analyse des décisions

## Integral Simplex Using Decomposition

#### **Presented by:**

Issmail El Hallaoui

Joint Work With :

Abdelouahab Zaghrouti

François Soumis

**International Workshop on Column Generation 2012** 

Contents	What/why?	The set partitioning problem
		The story of integral simplex
		The decomposition
	How?	The integral simplex using decomposition algorithm
		Properties
	So what?	Numerical results
		Open question and conclusion

Hummingbird

NY A

X

21.

Kinghabay

Mar Series V

the loss Rade

Me Bran I Mar Parels have 16 Sections

vptinple

والعلوانا

in the

West

150.00

Volute

Selin

in Sala

Sec. Sec. 1

a bart

Sugar al an in

Siste

23.00

NZ.

Wendy & Richard

All Larger

Goldfinch

Me Smar Center

Methon Consider

Winjhiliy

Mechanisty.

W-Jane Eline

Min Eller un Minla

M-Oder Cetyle

HalanaSapa

M-Chifarm Ma Jan Man Inst

> Dove White

ciest

Me Tim Copleilons

Mrs Jenny Hall Mr Shaari Mackellar

Miss Susan Keill

Garil of Min Sunas Keill

Min Sally Thorn My Martin Shippen Mina Julie Dormer

> Mr Gareth Court Miss Sue James Mr Leo Pianet Miss Amelie Pianet Mrs Olizia Pianet Mr Paul Whisbread Mrs Jeni Whisbread Miss Ali Haffilose

> > Skylark Mr Marcus Beand Mr Oli Beard Miss Miss Beard

http://www.coin-or.org/PuLP/CaseStudies/a\_set\_partitioning\_problem.html

### The set partitioning problem (SPP) Happy example: wedding

Let G be the set of guests. Each guest must be assigned a table. The tables are "feasible" subsets of G. We wish to maximize the total *happiness* of all of the tables.

#### The set partitioning problem (SPP) Mathematical formulation



Usually solved by branch and bound and cut and price and ...

## The set partitioning problem (SPP) Applications

- Weddings all over the world
- Transportation



Pairings



Bus driver scheduling



Locomotives



Trucking

• Clustering: data clustering, sensor clustering, ...



• The story began in 1969 with Trubin when he observed that the polytope Q of SPP is *quasi-integral.* 

**Meaning that:** every edge of conv(Q') is also an edge of Q where Q' is the set of its integer points.



## The story

• Interpretation: there exists a path from  $x_0$  to  $x^*$  where vertices  $x_i$  are all integer.



• **Trubin** (1969) proposed a variant of the simplex algorithm to solve this kind of problems.

## The story

#### Balas and Padberg quoted in 1972 criticizing Trubin:

Trubin, using a completely different line of reasoning, shows that all edges of the convex hull of the feasible integer solutions to Q are also edges of the feasible set of Q'. This property (interesting in itself) then implies the existence of a path containing only integer vertices between any two integer vertices of the feasible set. However, neither the upper bound on the number of pivots required to get from one integer vertex to the other nor the existence of a minimum-length path whose associated objective function values form a monotonic sequence follows directly from Trubin's result.



- Balas and Padberg (1972) proved the existence of a decreasing sequence of integer solutions leading to the optimal solution with at most *m* pivots.
- But in practice...



## The story

- Yemelichev, Kovalev and Kravtsov referred in 1984 (translation) to the algorithm as Integral Simplex for the first time.
- Thompson used in 2002 the integral simplex to solve instances with up to **163** constraints.
- Rönnberg and Larsson developed in 2009 an extension of the integral simplex method to the column generation context.





Example:  $G = \{g1, g2, g3, g4, g5, g6\}$ 

- A4 = A2 U A3
- A4 is compatible of lower cost 4 ( < 2 + 100)



Example:  $G = \{g1, g2, g3, g4, g5, g6\}$ 

**Definition:** if Ak is union of some columns of S, Ak is said to be compatible with S.

#### The reduced problem:

contains columns *compatible* with the partition



The reduced problem improves the current integer solution in polynomial time.



#### How to escape the local optimum?

 Balas and Padberg branch or pivot on negative coefficients (degenerate variables) !!!



- A5 U A6 = A1 U A4
- The combination  $C_1 = \{A5, A6\}$  is compatible and minimal.



#### **Definition:**

- A combination C<sub>k</sub> is said to be compatible if it is union of some columns of S.
- C<sub>k</sub> is minimal if it becomes incompatible by removing any column from it.



#### How could we find such combinations?

#### The complementary problem (CP):

contains columns *incompatible* with the partition

Mean reduced cost

Minimize  $\, ar{ar{c}} \, x$ 

Compatibility Matrix  $\longrightarrow M_X = 0$ C<sub>k</sub>  $\in$  solution subspace

Normalization constraint to close the cone

 $x_j \ge 0$ 

 $\sum_{j} x_{j} = 1 \quad <$ 

 $A_j$  such that  $x_j > 0$  are **disjoint** 

# Integral simplex using decomposition algorithm (ISUD)



#### **Properties** DW decomposition



**Proposition:** ISUD is a Dantzig-Wolfe decomposition of the set partitioning problem with  $Z^* = Z_{DW}$ 

#### **Properties Convergence**

Theorem: ISUD is exact and guarantees a decreasing sequence of integer solutions leading to the optimal solution.



No degeneracy, no pivoting on negative coefficients

## **Properties** Minimal combinations

- $\boldsymbol{S_0}$  : Set of columns of initial solution
- $S^*$ : Set of columns of optimal solution
- $C^*$ : Optimal combination ( $C^* = S^* S_0$ )



#### **Proposition**

- $C^* = \bigcup_{i=1}^k C_i$  is union of minimal combinations
- The complementary problem (CP) finds minimal combinations

#### **ISUD** is intrinsically parallelizable

#### **Properties** Integrality of combinations



## **ISUD favors integrality**

#### **Properties** Local cutting



#### **Easy handling and less cutting**

## **Properties** Local improvement



## The know how developed in metaheuristics could be recycled here



## **Numerical results**

- Tests on large instances up to 1600 constraints (instead of 163 of Thompson 2002) and 500 000 variables.
- The complementary problem often finds combinations of
  - disjoint columns (50%-90%)
  - small size (in average <= 10 columns/combination)</p>
- ISUD finds optimal solutions in 75% of cases within 20 minutes.
  - CPLEX takes 10 hours on the easiest large instance (gap 0)
  - CPLEX finds no feasible solutions for the hardest ones

## **Open question**

#### It's all about...



## **Open question**



The degree of incompatibility of a variable depends on M.

Question: could we find a compatibility matrix allowing to generate optimal disjoint combinations in polynomial time?

## Conclusion

- Proof of concept showing high potential
- Ongoing projects:
  - Extensive experimentation and refinement
  - Local cuts for set partitioning problem
  - Parallel version of ISUD
  - ISUD with cost projection



#### The story continues...

