

Université Toulouse III - Paul Sabatier

Report

Optimizing the acquisition plan of an earth optical observation satellite



Binôme :

- HADDOU Khalid : khalidhaddou99@gmail.com
- KEBIRI Issam Eddine : kebiri.isam.dine@gmail.com

Responsable pédagogique :

- Hélène FARGIER, Michel LEMAITRE, Gérard VERFAILLIE

Plan :

1. INTRODUCTION	2
2. DESCRIPTION DES EXPERIENCES.....	2
2.1. VARIABLES DE L'EXPERIENCES.....	2
2.2. MISE EN PLACE DES CONTRAINTES.....	3
2.3. FONCTION OBJECTIVE	4
3. RÉSULTATS.....	4
3.1. SOLUTION POUR LE JEU DE DONNEES <i>spotProba4.dat</i>	4
3.2. SOLUTION POUR LE JEU DE DONNEES <i>spotProba5.dat</i>	7
4. CONCLUSION	11

Figures :

Figure 1 : Run_spotProba4	5
Figure 2 : variables de décision assignedTo spotProba4.dat	5
Figure 3 : variables de décision probalInstrumentOk spotProba4.dat.....	6
Figure 4 : variables de décision selection spotProba4.dat	6
Figure 5 : Run_spotProba5	7
Figure 6 : variables de décision assignedTo spotProba5.dat	8
Figure 7 : variables de décision probalInstrumentOk spotProba5.dat.....	9
Figure 8 : variables de décision selection spotProba5.dat	9

1. INTRODUCTION

Ce rapport a pour objective de détailler les différentes étapes pour déterminer les meilleurs plans d'acquisition d'images par instruments : quel ordre d'images sera acquis et par quel instrument en respectant un certain nombre de contraintes.

Pour réaliser ce TP, nous construirons un modèle du problème basé sur la programmation linéaire et la programmation par contraintes, ce modèle est réalisé grâce à l'environnement "OPL IDE" et au langage de script qui appelle les bibliothèques ILOG CP en se basant sur les données *spotproba1.dat*, *spotproba2.dat*, *spotproba3.dat*, *spotproba4.dat* et *spotproba5.dat*.

2. DESCRIPTION DES EXPÉRIENCES

Dans cette partie, nous allons présenter la description des expériences, les résultats et les discussions.

2.1.VARIABLES DE L'EXPÉRIENCE

Les variables étant donné dans la fiche de TP, nous allons les retranscrire pour une meilleure compréhension de ce rapport

```
// nbr img
int NbImages = ... ;
// boucle pour parcourir les images
range Images = 1 .. NbImages ;
// types img: 1 for mono, 2 for stereo
int TY[Images] = ... ;
// mémoire requise pour chaque img
int PM[Images] = ... ;
// prix de chaque img
int PA[Images] = ... ;
// probability inf of Cloudy Weather
float ProbaInf[Images] = ... ;
// probability sup of Cloudy Weather
float ProbaSup[Images] = ... ;
// nbr d'instruments
int NbInstruments = ... ;
// boucle pour parcourir les instruments
range Instruments = 1 .. NbInstruments ;
// probability of failure of each instrument
float Failure[Instruments] = ... ;
// starting time of each img, depending on the instrument
int DD[Images, Instruments] = ... ;
// depointing angle of each img, depending on the instrument
int AN[Images, Instruments] = ... ;
```

```
// duration of acquisition, common to all the img
int DU = ... ;
// rotation speed of the instruments
int VI = ... ;
// Capacity of the memory
int PMmax = ... ;
range bool = 0 .. 1 ;
{int} PossibleStartDates = {DD[ima, ins] | ima in Images, ins in Instruments};
// img i est sélectionné et capturé
dvar int selection[Images] in bool ;
// img i est assigné à l'instrument j
dvar int assignedTo[Images, Instruments] in bool ;
// probabilité que l'image ima sera correctement acquise par l'instrument qui lui est assigné
dvar float probaInstrumentOK[Images] in 0..1 ;
```

2.2.MISE EN PLACE DES CONTRAINTES

Contrainte 01 : pour chaque instrument, si, sur cet instrument, le temps de transition entre la fin d'une image et le début d'une autre n'est pas suffisant, il n'est pas possible de réaliser les deux manipulations sur cet instrument

```
forall(ordered ima1, ima2 in Images, ins in Instruments :
abs(DD[ima1,ins] - DD[ima2,ins]) * VI < DU * VI + abs(AN[ima1,ins] -
AN[ima2,ins]) )
{
    assignedTo[ima1,ins] + assignedTo[ima2,ins] <= 1 ;
} ;
forall(ordered ima1, ima2 in Images, ins in Instruments :
abs(DD[ima1,ins] - DD[ima2,ins]) < DU)
{
    assignedTo[ima1,ins] + assignedTo[ima2,ins] <= 1 ;
} ;
```

Contrainte 02 : quantité d'images qui peut être enregistré

```
sum(ima in Images) (PM[ima] * selection[ima]) <= PMmax ;
```

Contrainte 03 : les images stéréo sont réalisées par les instruments 1 et 3, ou pas réalisé du tout

```
forall(ima in Images: TY[ima] == 2) {
    assignedTo[ima, 1] == 1 ;
    assignedTo[ima, 3] == 1 ;
}
```

Contrainte 04 : assignation d'instrument que pour les images sélectionnées

```
forall(ima in Images) {
sum(ins in Instruments) assignedTo[ima, ins] == selection[ima] * TY[ima] ;
}
```

Contrainte 05 : probabilité d'échec de l'instrument stéréo

```
forall (ima in Images :TY[ima]==2) {
probaInstrumentOK[ima] == (1-Failure[1])*(1-Failure[3]) ;
}
```

Contrainte 06 : probabilité d'échec de chaque instrument

```
forall (ima in Images :TY[ima]==1) {
probaInstrumentOK[ima] == sum(ins in Instruments) (assignedTo[ima,ins]*(1-
Failure[ins])) ;
}
```

2.3.FONCTION OBJECTIVE

La fonction objective consiste à maximiser la somme des prix des photos (payof) :

On procédera avec un parcours des images en multipliant le prix de chaque image avec un booléen (1 si l'image est sélectionnée et capturée, 0 sinon) multiplié par la probabilité que l'image soit acquise correctement par l'instrument qui lui est assigné (probabilité entre 0 et 1) multiplié par 1 moins la probabilité supérieure que la météo soit nuageuse.

```
// maximize the sum of the payoffs : fonction objective
maximize sum(ima in Images) (PA[ima] * selection[ima] * probaInstrumentOK[ima]
* (1 - ProbaSup[ima])) ;
```

3. RÉSULTATS

3.1.SOLUTION POUR LE JEU DE DONNÉES *spotProba4.dat*

Après ajustement de notre modèle sur les jeux de données *spotProba1.dat*, *spotProba2.dat* et *spotProba3.dat*, nous l'avons appliqué au jeu de données *spotProba4.dat* qui contient 20 images et auquel la probabilité d'échec de chaque instrument est égale à zéro, le fichier de configuration *Run_spotProba4* qui contient le fichier de données et notre modèle à donner les résultats suivants :

olution avec l'objectif 165

Nom	Valeur
Données (16)	
AN	[[10 10 10] [-5 0 -5] [20 20 20] [-10 0 -10] [5 5 5] [20 20 20] [10 10 10] [-10 0 -10] [5 5 5] [15 15 15] [0 0 0] [-5 0 -5] [20 20 20] [-15 0 -15] [5 5 5] [20 20 20] [15 15 15] [-10 0 -10] [5 5 5] [-15 -15 -15]]
bool	0.1
DD	[[100 200 300] [130 0 330] [150 250 350] [200 0 400] [220 320 420] [230 330 430] [250 350 450] [260 0 460] [290 390 490] [300 400 500] [300 400 500] [320 0 520] [330 430 530] [400 0 600] [420 520 620] [440 0 740]]
DU	20
Failure	[0 0 0]
Images	1..20
Instruments	1..3
NbImages	20
NbInstruments	3
PA	[10 20 40 30 10 20 40 20 50 20 10 20 50 30 20 10 40 10 30 20]
PM	[10 20 10 20 10 10 20 10 10 20 10 20 10 10 10 20 10 10]
PMmax	150
ProbaInf	[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
ProbaSup	[0.1 0.7 0.1 0.7 0.1 0.1 0.1 0.7 0.1 0.1 0.1 0.7 0.1 0.1 0.1 0.7 0.1 0.1]
TY	[1 2 1 2 1 1 2 1 1 2 1 2 1 1 2 1 1]
VI	1
Variables de décision (3)	
assignedTo	[[0 0 0] [1 0 1] [0 1 0] [1 0 1] [0 0 0] [0 0 0] [0 0 0] [1 0 1] [0 1 0] [0 0 0] [0 0 0] [1 0 1] [0 1 0] [1 0 1] [0 0 0] [0 0 0] [0 0 0] [1 0 1] [0 0 0] [0 0 0]]
probaInstrumentOK	[0 1 1 0 0 0 1 1 0 0 1 1 0 0 0 1 0 0]
selection	[0 1 1 0 0 0 1 1 0 0 1 1 0 0 0 1 0 0]

Figure 1 : Run_spotProba4

Solution optimale avec objective 165

Les variables de décision :

- 1- assignedTo : l'image i est assignée à l'instrument j, par exemple l'image 9 est assignée à l'instrument 2

Images (taille 20)	Instruments (taille 3)		
	1	2	3
1	0	0	0
2	1	0	1
3	0	1	0
4	1	0	1
5	0	0	0
6	0	0	0
7	0	0	0
8	1	0	1
9	0	1	0
10	0	0	0
11	0	0	0
12	1	0	1
13	0	1	0
14	1	0	1
15	0	0	0
16	0	0	0
17	0	0	0
18	1	0	1
19	0	0	0
20	0	0	0

Figure 2 : variables de décision assignedTo spotProba4.dat

- 2- `probaInstrumentOk` : probabilité que l'image i soit correctement acquise par l'instrument qui lui est assigné, par exemple l'image 12 à la probabilité 1 (donc sûre) qu'elle soit acquise correctement

Images (taille 20)	Valeur
1	0
2	1
3	1
4	1
5	0
6	0
7	0
8	1
9	1
10	0
11	0
12	1
13	1
14	1
15	0
16	0
17	0
18	1
19	0
20	0

Figure 3 : variables de décision `probaInstrumentOk` `spotProba4.dat`

- 3- `selection` : l'image i est sélectionné et capturé, par exemple l'image 15 ne sera pas sélectionné et capturé

Images (taille 20)	Valeur
1	0
2	1
3	1
4	1
5	0
6	0
7	0
8	1
9	1
10	0
11	0
12	1
13	1
14	1
15	0
16	0
17	0
18	1
19	0
20	0

Figure 4 : variables de décision `selection` `spotProba4.dat`

Solution optimale avec objective 158,8

Les variables de décision :

- 1- assignedTo : l'image i est assignée à l'instrument j, par exemple l'image 21 est assignée à l'instrument 1 et 3

Images (taille 40)	Instruments (taille 3)		
	1	2	3
1	0	0	0
2	1	0	1
3	0	0	0
4	1	0	1
5	0	0	0
6	0	0	0
7	0	0	0
8	1	0	1
9	0	1	0
10	0	0	0
11	0	0	0
12	1	0	1
13	0	1	0
14	1	0	1
15	0	0	0
16	0	0	0
17	0	0	0
18	1	0	1
19	0	0	0
20	0	0	0
21	1	0	1
22	1	0	1
23	0	0	0
24	1	0	1
25	0	0	0
26	0	0	0
27	0	0	0
28	1	0	1
29	0	0	0
30	0	0	0
31	0	0	0
32	1	0	1
33	0	0	0
34	1	0	1
35	0	0	0
36	0	0	0
37	0	0	0
38	1	0	1
39	0	0	0
40	1	0	1

Figure 6 : variables de décision assignedTo spotProba5.dat

- 2- `probalInstrumentOk` : probabilité que l'image *i* soit correctement acquise par l'instrument qui lui est assigné, par exemple l'image 12 à la probabilité 0,29 qu'elle soit acquise correctement

Images (taille 40)	Valeur		
1	0	21	0.2997
2	0.2997	22	0.2997
3	0	23	0
4	0.2997	24	0.2997
5	0	25	0
6	0	26	0
7	0	27	0
8	0.2997	28	0.2997
9	0.999	29	0
10	0	30	0
11	0	31	0
12	0.2997	32	0.2997
13	0.999	33	0
14	0.2997	34	0.2997
15	0	35	0
16	0	36	0
17	0	37	0
18	0.2997	38	0.2997
19	0	39	0
20	0	40	0.2997

Figure 7 : variables de décision `probalInstrumentOk` `spotProba5.dat`

- 3- `selection` : l'image *i* est sélectionnée et capturée, par exemple l'image 15 ne sera pas sélectionnée et capturée

Images (taille 40)	Valeur		
1	0	21	1
2	1	22	1
3	0	23	0
4	1	24	1
5	0	25	0
6	0	26	0
7	0	27	0
8	1	28	1
9	1	29	0
10	0	30	0
11	0	31	0
12	1	32	1
13	1	33	0
14	1	34	1
15	0	35	0
16	0	36	0
17	0	37	0
18	1	38	1
19	0	39	0
20	0	40	1

Figure 8 : variables de décision `selection` `spotProba5.dat`

Et pour finir la fonction à maximiser :

```
Maximize
obj1: [ 18 selection(1) * probaInstrumentOK(1)
+ 36 selection(2) * probaInstrumentOK(2)
+ 72 selection(3) * probaInstrumentOK(3)
+ 54 selection(4) * probaInstrumentOK(4)
+ 18 selection(5) * probaInstrumentOK(5)
+ 36 selection(6) * probaInstrumentOK(6)
+ 72 selection(7) * probaInstrumentOK(7)
+ 36 selection(8) * probaInstrumentOK(8)
+ 90 selection(9) * probaInstrumentOK(9)
+ 36 selection(10) * probaInstrumentOK(10)
+ 18 selection(11) * probaInstrumentOK(11)
+ 36 selection(12) * probaInstrumentOK(12)
+ 90 selection(13) * probaInstrumentOK(13)
+ 54 selection(14) * probaInstrumentOK(14)
+ 36 selection(15) * probaInstrumentOK(15)
+ 18 selection(16) * probaInstrumentOK(16)
+ 72 selection(17) * probaInstrumentOK(17)
+ 14 selection(18) * probaInstrumentOK(18)
+ 42 selection(19) * probaInstrumentOK(19)
+ 28 selection(20) * probaInstrumentOK(20)
+ 56 selection(21) * probaInstrumentOK(21)
+ 28 selection(22) * probaInstrumentOK(22)
+ 56 selection(23) * probaInstrumentOK(23)
+ 42 selection(24) * probaInstrumentOK(24)
+ 84 selection(25) * probaInstrumentOK(25)
+ 24 selection(26) * probaInstrumentOK(26)
+ 48 selection(27) * probaInstrumentOK(27)
+ 24 selection(28) * probaInstrumentOK(28)
+ 60 selection(29) * probaInstrumentOK(29)
+ 48 selection(30) * probaInstrumentOK(30)
+ 12 selection(31) * probaInstrumentOK(31)
+ 20 selection(32) * probaInstrumentOK(32)
+ 50 selection(33) * probaInstrumentOK(33)
+ 30 selection(34) * probaInstrumentOK(34)
+ 20 selection(35) * probaInstrumentOK(35)
+ 10 selection(36) * probaInstrumentOK(36)
+ 24 selection(37) * probaInstrumentOK(37)
+ 6 selection(38) * probaInstrumentOK(38)
+ 18 selection(39) * probaInstrumentOK(39)
+ 24 selection(40) * probaInstrumentOK(40) ] / 2
```

4. CONCLUSION

Le modèle *Optimizing_the_acquisition_plan.mod* (avec la fonction de maximisation du cout (payof) des images et la sélection et capture ou non des images par les trois instruments ainsi qu'aux calculs de probabilités qu'une image soit acquise correctement) que nous avons implémenté à apporter des résultats convaincants aux deux jeux de données *spotProba4.dat* et *spotProba5.dat* en respectant les contraintes indiquées dans la fiche TP.