Challenge ROADEF' 2005
Problème du Car Sequencing

Alain NGUYEN

RENAULT - alain.nguyen@renault.com

1-DESCRIPTION DU PROBLÈME 3
1.1-CONTEXTE 3
1.1.1 - Processus de planification/ordonnancement 3
1.1.2 - Besoins de l'atelier peinture 3
1.1.3 - Besoins de l'atelier montage 3
1.1.4 – Rôle de la journée J-1 4
1.2-PROBLÈME À RÉSOUDRE 4
1.2.1 – Optimisation multi-critères pour le cas « contraintes de ratio en priorité » 5
1.2.2 – Optimisation multi-critères pour le cas « rafales de peinture en priorité » 5
1.2.3 – Précisions sur la comptabilisation des violations de contraintes de ratio 5
1.2.4 –Contraintes de ratio prioritaires faciles/difficiles 7
2 - MODALITÉS PRATIQUES 7
2.1 - JEUX D'INSTANCES FOURNIES PAR RENAULT 7
2.1.1 – Les bases d'instances 8
2.1.2 – Contenu de chaque instance 8
2.1.2 – Format des fichiers 8
2.1.2.1 Fichier de la longueur maxi de rafales de peinture 9
2.1.2.2 Fichier des objectifs d'optimisation 9
2.1.2.3 Fichier des contraintes de ratio 10
2.1.2.4 Fichier des véhicules 11
2.2 – FOURNITURES ATTENDUES DES CANDIDATS 12
2.2.1-Dossier de réponse 12
2.2.2 - Fichiers de solutions 13
2.2.3 - Programmes 13
2.2.4 - Organisation de l'arborescence des dossiers candidats 13
2.3 - PROCÉDURE D'ÉVALUATION ET DE CLASSEMENT 14
2.3.1 – Temps imparti et plateforme logicielle/matérielle 14
2.3.2 – Programme d'évaluation 14
2.3.3 - Notation sur une instance de problème 14
2.3.3.1 – Cas d'une instance avec « contraintes de ratio en priorité » 15
2.3.3.2 – Cas d'une instance avec « rafales de peinture en priorité » 15
2.3.4 - Notation globale sur l'ensemble des instances de problème d'une base 16
2.4 – PLANNING DU CHALLENGE 16

1-Description du problème

1.1-Contexte

1.1.1 - Processus de planification/ordonnancement

Quotidiennement, les commandes clientes sont transmises en temps réel dans les usines de carrosserie-montage (usines d'assemblage de véhicules).

La tâche quotidienne des usines est (1) d'affecter une journée de fabrication à chaque véhicule commandé en fonction de contraintes capacitaires des lignes de fabrication et des délais promis aux clients. Il s'agit ensuite (2) d'ordonnancer les véhicules à l'intérieur de chaque journée de fabrication, en satisfaisant au mieux les besoins respectifs des ateliers de la ligne de fabrication : ateliers de tôlerie, peinture et montage. La séquence de véhicules calculée constitue le **« film »** qui sera envoyé à la fabrication.

On se focalisera sur les besoins des ateliers de peinture et de montage, considérant que l'atelier de tôlerie ne génère pas d'impératifs sur l'ordonnancement de chaque journée.

Lors de l'ordonnancement de chaque journée, on ne remet plus en cause la composition des journées décidée à l'étape (1).

Une application opérationnelle gère ce processus de planification/ordonnancement, en utilisant la Programmation Linéaire pour l'étape (1) et le recuit simulé pour l'étape (2).

1.1.2 - Besoins de l'atelier peinture

Le besoin de l'atelier peinture est de minimiser la consommation de solvant, lequel solvant est utilisé pour purger les pistolets de peinture à chaque changement de teinte dans le film véhicules.

On cherche donc à « grouper » les véhicules de même teinte, ce qui revient à minimiser le nombre de purges, c-a-d le nombre de changements de teintes dans le film, ce qui revient encore à générer les rafales de peinture les plus longues possibles.

Mais chaque rafale de teintes ne doit pas dépasser une longueur maximale, car il faut périodiquement purger les pistolets de peinture, même s'il n'y a pas eu de changements de teinte. C'est une **contrainte dure**.

1.1.3 - Besoins de l'atelier montage

Afin de lisser la charge de travail sur les différents postes en bord de chaîne, on cherche à « écarter » les véhicules « difficiles », c-a-d ayant des équipements nécessitant des opérations lourdes. En d'autres termes, on cherche à limiter la densité des véhicules difficiles pour ne pas surcharger les postes de travail concernés par l'assemblage de ces véhicules.

Ce besoin d'écartement est formalisé par une **contrainte de ratio N/P**. La contrainte porte sur une caractéristique technique qui nécessite des opérations lourdes (par exemple, le toit ouvrant, la climatisation, etc).

La contrainte de ratio N/P signifie qu'on ne doit pas trouver une densité supérieure à N/P de véhicules concernés par la contrainte de ratio. Par exemple, si N/P = 3/5, on ne doit pas trouver plus de 3 véhicules concernés par la contrainte de ratio dans toute séquence de 5 véhicules dans le film.

Lorsque N=1, cela signifie que 2 véhicules concernés par la contrainte de ratio 1/P doivent être séparés par au moins P-1 positions dans le film. Exemple : contrainte de ratio 1/5 : X_ _ _ _X, X=véhicule concerné par la contrainte de ratio.

On définit 2 catégories de contraintes de ratio, les prioritaires et les non prioritaires. Les contraintes de ratio prioritaires correspondent à des contraintes « lourdes » de l'atelier de montage, les non prioritaires étant associés à des opérations plus légères. On veut ainsi privilégier la satisfaction des contraintes de ratio prioritaires sur les contraintes de ratio non prioritaires. En d'autres termes, on veut éviter que le respect d'un ratio associé à une opération légère se fasse au détriment d'un ratio associé à une opération lourde.

Les contraintes de ratio sont des **contraintes molles** : on ne peut pas savoir a priori si l'ensemble des contraintes de ratio peut être respecté lorsqu'on ordonnance une journée de fabrication. L'objectif de l'optimisation consiste donc à minimiser le nombre de violations des contraintes de ratio.

1.1.4 – Rôle de la journée J-1

Lorsqu'on ordonnance les véhicules de la journée J, on doit tenir compte des derniers véhicules de la journée J-1. On considère que ces véhicules sont déjà séquencés et que leurs positions sont **figées** pour le calcul d'ordonnancement de la journée J. Il faut donc tenir compte des derniers véhicules de la journée J-1 pour compter les violations de contraintes de ratio sur la journée J. En revanche, on ne tiendra pas compte de la journée J+1.

1.2-Problème à résoudre

Il s'agit de construire un film véhicules satisfaisant les besoins de l'atelier peinture et ceux de l'atelier montage. On peut adopter 2 stratégies possibles selon les usines :
- « **contraintes de ratio en priorité** » : les besoins de l'atelier montage sont prioritaires par rapport à ceux de l'atelier peinture
- « **rafales de peinture en priorité** » : les besoins de l'atelier peinture sont prioritaires par rapport à ceux de l'atelier montage.

1.2.1 – Optimisation multi-critères pour le cas « contraintes de ratio en priorité »

On cherche à optimiser les 3 critères suivants, du plus prioritaire au moins prioritaire, et **sans compensation** entre les critères, pour construire le film véhicules :

1. minimiser le nombre de violations des contraintes de ratio prioritaires
2. minimiser le nombre de purges de peinture
3. minimiser le nombre de violations des contraintes de ratio non prioritaires

ou

1. **minimiser le nombre de violations des contraintes de ratio prioritaires**
2. **minimiser le nombre de violations des contraintes de ratio non prioritaires**
3. **minimiser le nombre de purges de peinture**

ou

1. **minimiser le nombre de violations des contraintes de ratio prioritaires**
2. **minimiser le nombre de purges de peinture**

Certaines usines ne déclarent pas de contraintes de ratio non prioritaires.

1.2.2 – Optimisation multi-critères pour le cas « rafales de peinture en priorité »

On cherche à optimiser les 3 critères suivants, du plus prioritaire au moins prioritaire, et **sans compensation** entre les critères, pour construire le film véhicules :

(1) minimiser le nombre de purges de peinture
(2) minimiser le nombre de violations des contraintes de ratio prioritaires
(3) minimiser le nombre de violations des contraintes de ratio non prioritaires

ou

(1) minimiser le nombre de purges de peinture
(2) minimiser le nombre de violations des contraintes de ratio prioritaires

Pour les deux optimisations, la seule contrainte « dure » du problème est constituée par le respect de la longueur maximale des rafales de peinture.

Il ne doit pas y avoir compensation entre les objectifs : la résolution de l'objectif de rang O+1 ne doit pas dégrader le résultat de l'objectif de rang O.

1.2.3 – Précisions sur la comptabilisation des violations de contraintes de ratio

Pour chaque contrainte de ratio N/P, on cherche à obtenir la configuration idéale suivante : sur toute fenêtre glissante de taille P sur le film, on ne doit pas trouver plus de N véhicules concernés par la contrainte de ratio. Dans le cas d'une contrainte de ratio du type 1/P, on vise

un film où 2 véhicules dans le film et concernés par la contrainte de ratio sont distants d'au moins P-1 positions : X _ _ _ X pour un ratio de 1/4.

Si on ne parvient pas à obtenir zéro violations sur cette contrainte de ratio, on cherche à « écarter » au maximum les véhicules concernés par la contrainte. Il s'agit de lisser au mieux la charge de travail sur les postes de travail contraints, c-a-d les postes à l'origine d'une contrainte de ratio.

On essaie d'obtenir cet « écartement» en comptabilisant les contraintes de ratio sur des fenêtres glissantes sur tout le film. Moins les véhicules seront écartés et plus nombreuses seront les violations, car la violation avec des véhicules moins écartés sera comptabilisée dans davantage de fenêtres glissantes, et le nombre de violations pourra être plus élevé dans chaque fenêtre glissante.

Exemple : contrainte de ratio de 1/5 => on comptabilise les violations avec des fenêtres glissantes de taille 5
_ _ _ X _ _ _ X : 1 violation sur la fenêtre glissante _ _ _ X _ _ _ X
_ _ _ X _ _X _ : 2 violations sur les fenêtres glissantes _ _ _ X _ _X _ et _ _ _ X _ _X _

Nombre de violations comptabilisées sur une fenêtre glissante = (nombre de véhicules de la fenêtre glissante concernés par le ratio) – (numérateur du ratio) **si (nombre de véhicules de la fenêtre glissante concernés par le ratio) > (numérateur du ratio)** **= 0 sinon**

Cette méthode de comptabilisation s'applique à la fois aux contraintes de ratio 1/P et N/P (N<>1).

Exemple : contrainte de ratio de 1/5
_ _ X X _ _ X _ : on comptabilise 2 violations sur cette fenêtre glissante : (3 véhicules concernés par la contrainte de ratio) – 1 (= numérateur du ratio)

On ne cherche pas à écarter les véhicules au-delà de la contrainte de ratio. Si les véhicules satisfont à la contrainte de ratio de 1/5 de l'exemple, on ne cherchera pas à les écarter davantage.

L'objectif consistera donc à minimiser le nombre de violations des contraintes de ratio sur l'ensemble des fenêtres glissantes sur le film, et ce pour l'ensemble des contraintes de ratio.

Comme on doit tenir compte des véhicules déjà séquencés de la journée J-1 (si on est en train d'ordonnancer la journée J), les fenêtres glissantes doivent débuter dans la journée J-1. Pour un ratio N/P, la 1ère fenêtre glissante contient les P-1 derniers véhicules (dans l'ordre de la séquence figée) de la journée J-1 et le 1er véhicule de la journée J. Pour rappel, les positions des véhicules de la journée J-1 sont figées.

Exemple : 1ère fenêtre glissante pour un ratio 1/5
XXXXXYYYYY
X : derniers véhicules de la journée J-1
Y : premiers véhicules de la journée J

On ne tient pas compte des véhicules de la journée J+1 quand on séquence la journée J, mais pour chaque contrainte de ratio, on veut comptabiliser les violations sur les derniers véhicules de la journée J, en faisant comme si les premiers véhicules de la journée J+1 ne sont pas concernés par la contrainte.

Pour un ratio N/P, les dernières fenêtres glissantes auront une longueur comprise entre P-1 et (N+1). Si dans une telle fenêtre, le nombre de véhicules associés au ratio est strictement supérieur à N, alors il y a des violations, quelque soient les véhicules positionnés 000au début de la journée J+1.

Cette méthode de comptabilisation permet de pénaliser les violations sur les derniers P-1 véhicules de la journée J (pour un écart N/P) autant que celles se produisant sur les véhicules antérieurs.

Exemple : les dernières fenêtres glissantes pour un ratio 1/5
YYYYYYYY
YYYYYYYY
YYYYYYYY
Y : derniers véhicules de la journée J dans l'ordre de la séquence ordonnancée

1.2.4 –Contraintes de ratio prioritaires faciles/difficiles

RENAULT fournira des instances de problèmes avec des contraintes de ratio prioritaires « faciles » ou « difficiles ». Cela aura une influence sur la notation des équipes. On ne fait pas cette distinction sur les contraintes de ratio non prioritaires.

Une instance de problème est considérée comme ayant des contraintes de ratio prioritaires **faciles** si l'application opérationnelle de RENAULT parvient à construire une séquence ne produisant aucune violation sur l'ensemble des contraintes de ratio prioritaires. Les contraintes de ratio prioritaires faciles sont donc réalisables.

Une instance de problème est considérée comme ayant des contraintes de ratio prioritaires **difficiles** si l'application opérationnelle de RENAULT ne parvient pas à construire de séquence ne produisant aucune violation sur l'ensemble des contraintes de ratio prioritaires. C'est le cas notamment des instances avec des ratios supérieurs au mix des véhicules : par exemple, la journée à ordonnancer contient 25% de véhicules d'une certaine caractéristique technique et on impose une contrainte de ratio prioritaire de 1/5.
Les contraintes de ratio prioritaires difficiles peuvent être réalisables ou non réalisables.

2 - Modalités pratiques

2.1 - Jeux d'instances fournies par RENAULT

2.1.1 – Les bases d'instances

RENAULT fournira progressivement trois bases d'instances de problèmes, ceci dans le souci de donner davantage de possibilités aux candidats de mieux régler leurs programmes :

- BASE A : base qui est disponible dès le début du challenge. Elle sert à la fois aux candidats pour régler leurs algorithmes et au jury pour la sélection des candidats pour la finale. C'est la **base de la phase de qualification**.
- BASE B : base qui sera fournie aux candidats finalistes afin d'affiner encore plus leurs algorithmes.
- BASE X : base qui sera utilisée pour le classement final des finalistes avant le congrès ROADEF'2005. C'est la **base de la phase finale**, elle ne sera divulguée aux candidats qu'après l'annonce du classement final au congrès ROADEF'2005.

Le travail de réglage des algorithmes sur plusieurs bases devrait améliorer la robustesse des algorithmes face à la diversité des données. Cette robustesse est extrêmement importante en opérationnel. Il n'est en effet pas réaliste de demander aux équipes de maintenance informatique de régler les algorithmes chaque fois que les données (contraintes de ratio, véhicules) changent significativement sur l'une des usines RENAULT.

2.1.2 – Contenu de chaque instance

Une liste d'instances de problèmes sera fournie, chaque instance correspondant au jeu de données d'une usine. La même usine peut être représentée dans plusieurs instances de problèmes.

Pour chaque instance de problème, on donnera :

- la liste des contraintes de ratio prioritaires
- la liste des contraintes de ratio non prioritaires
- la longueur maximale pour les rafales de peinture
- la journée J de véhicules à ordonnancer et une partie de la journée J-1 de véhicules figés (cette partie de la journée J-1 est composée de DenMax-1 véhicules, DenMax étant le max des dénominateurs des contraintes de ratio).
- Pour chaque véhicule (à séquencer ou figé), on fournira un identifiant, la journée de fabrication (date J-1 ou date J), la teinte et un flag (0/1) par contrainte de ratio indiquant si le véhicule est concerné par la contrainte de ratio, le numéro de séquence calculé par l'application opérationnelle de RENAULT.
- l'ordre des objectifs d'optimisation.

Certaines instances de problèmes peuvent ne pas contenir de contraintes de ratio non prioritaires. En effet, certaines usines peuvent ne pas en déclarer, et mettre tous les contraintes de ratio en prioritaire.

2.1.2 – Format des fichiers

Chaque instance se présente sous la forme d'un répertoire instance_XXXX contenant les fichiers suivants :

- Fichier de la longueur maxi de rafales de peinture : « *paint_batch_limit.txt* »
- Fichier des objectifs d'optimisation : « *optimization_objectives.txt* »
- Fichier des contraintes de ratio : « *ratios.txt* »

- Fichier des véhicules figés de la journée J-1 et des véhicules à ordonnancer de la journée J : « *vehicles.txt* »

Il s'agit de fichiers texte ascii utilisant le «; » comme séparateur de champs. Les noms de fichiers, ainsi que les libellés texte présents dans ces fichiers, sont en Anglais.

2.1.2.1 Fichier de la longueur maxi de rafales de peinture

Ce fichier contient un entier représentant la longueur maximale acceptée pour les rafales de peinture.

Exemple de fichier « *paint_batch_limit.txt* » :
limitation;
10;

2.1.2.2 Fichier des objectifs d'optimisation

Ce fichier donne l'ordre entre les différents objectifs d'optimisation, de l'objectif le plus prioritaire vers le moins prioritaire. On donne les couples « numero_ordre_objectif;libelle_objectif ». Dans le libellé des objectifs d'optimisation, on précise s'il s'agit de contraintes de ratio prioritaires « faciles » ou « difficiles ».

Différents cas de figure du fichier « *optimization_objectives.txt* » :

Cas des contraintes de ratio en priorité avec des contraintes de ratio prioritaires faciles

rank;objective name;
1;high_priority_level_and_easy_to_satisfy_ratio_constraints;
2;paint_color_batches;
3;low_priority_level_ratio_constraints;

ou

rank;objective name;
1;high_priority_level_and_easy_to_satisfy_ratio_constraints;
2;low_priority_level_ratio_constraints;
3;paint_color_batches;

ou

rank;objective name;
1;high_priority_level_and_easy_to_satisfy_ratio_constraints;
2;paint_color_batches;

Cas des rafales de peinture en priorité avec des contraintes de ratio prioritaires faciles

rank;objective name;
1;paint_color_batches;
2;high_priority_level_and_easy_to_satisfy_ratio_constraints;

3;low_priority_level_ratio_constraints;

ou

rank;objective name;
1;paint_color_batches;
2;high_priority_level_and_easy_to_satisfy_ratio_constraints;

Cas des contraintes de ratio en priorité avec des contraintes de ratio prioritaires difficiles

rank;objective name;
1 ;high_priority_level_and_difficult_to_satisfy_ratio_constraints;
2;paint_color_batches;
3;low_priority_level_ratio_constraints;

ou

rank;objective name;
1 ;high_priority_level_and_difficult_to_satisfy_ratio_constraints;
2;paint_color_batches;

ou

1;high_priority_level_and_difficult_to_satisfy_ratio_constraints;
2;low_priority_level_ratio_constraints;
3;paint_color_batches;

2.1.2.3 Fichier des contraintes de ratio

Ce fichier définit les contraintes de ratio prioritaires et non prioritaires. Pour chaque ratio, on donne le ratio N/P, le flag de priorité (1=prioritaire, 0=non prioritaire) et son identifiant (le rang du ratio).

Exemple de fichier « ratios.txt » :

Ratio Prio
; ; Ident;
1/2; 1; HPRC1;
1/8; 1; HPRC2;
2/3; 1; HPRC3;
1/4; 1; HPRC4;
3/4; 0; LPRC1;
1/3; 0; LPRC2;
2/3; 0; LPRC3;

Dans cet exemple, on fournit 4 contraintes de ratio prioritaires et 3 contraintes de ratio non prioritaires.

2.1.2.4 Fichier des véhicules

Ce fichier décrit les véhicules figés de la journée J-1 et les véhicules de la journée J qui sont à ordonnancer.
Pour chaque véhicule, on trouvera les champs suivants :

1. **Date** : la journée prévisionnelle de fabrication du véhicule, au format "AAAA SS J", où AAAA désigne l'année, SS la semaine industrielle dans l'année (SS=1..52), et J le jour dans la semaine (1 pour lundi, 2 pour mardi, etc.)
2. **SeqRank** : le rang du véhicule dans la journée. Il s'agit du rang calculé par l'application opérationnelle de RENAULT. La numérotation commence à 1.
3. **Ident** : identifiant du véhicule.
4. **Paint Color** : code couleur du véhicule, pour la problématique des rafales de teinte.
5. **HPRCi** : '1' si le véhicule matche le ratio prioritaire RPi, '0' sinon. Nombre de champs RPi = nombre de ratios prioritaires.
6. **LPRCi** : idem que **HPRCi** pour les ratios non prioritaires.

Sur la journée J-1 seront fournis uniquement les véhicules nécessaires au comptage des violations de contraintes de ratio de la journée J. Nombre de véhicules de la journée J-1 = (max des dénominateurs des contraintes de ratio) - 1.

Exemple de fichier « *vehicles.txt* » :

Date;SeqRank;Ident;Paint Color;HPRC1; HPRC2; HPRC3; HPRC4; LPRC1; LPRC2; LPRC3
2003 18 3;997;025031830079;5;1;0;0;0;0;0;0
2003 18 3;998;025031850339;2;0;0;0;0;1;0;0
2003 18 3;999;025031920802;1;1;0;1;1;1;1;1
2003 18 3;1000;025031830269;1;0;0;1;0;1;0
2003 18 3;1001;025031850094;2;0;0;0;0;0;0
2003 18 3;1002;025031910010;2;1;1;0;0;1;0
2003 18 3;1003;025031850332;2;0;0;0;0;1;0
2003 18 3;1004;025031820317;1;1;0;1;1;1;1
2003 18 5;1;025031910825;3;0;0;1;0;1;0;1;0
2003 18 5;2;025031820293;7;1;0;1;1;1;0;1;0
2003 18 5;3;025031850497;2;0;0;0;0;0;0;0;0
2003 18 5;4;025031911085;5;1;0;1;0;1;1;1;0
2003 18 5;5;025031850299;5;0;0;1;0;1;0;1;0
2003 18 5;6;025031850082;2;1;0;0;0;0;0;0;0
2003 18 5;7;025031850783;5;0;0;1;0;1;1;1;0
2003 18 5;8;025031920415;5;1;1;1;1;0;0;1;0
2003 18 5;9;025032010270;5;0;0;0;0;0;0;0;0
2003 18 5;10;025031911083;5;1;0;1;0;1;1;1
2003 18 5;11;025031910060;5;0;0;0;0;0;0;1
2003 18 5;12;025031970126;5;1;0;1;1;0;0;1
2003 18 5;13;025031910182;2;0;0;0;0;1;0;0
2003 18 5;14;025031850845;1;1;0;1;0;1;1;1
2003 18 5;15;025031910373;1;0;0;1;0;0;0;1
2003 18 5;16;025031850597;1;1;0;0;0;1;0;0
2003 18 5;17;025031830304;1;0;0;1;1;1;0;1

2003 18 5;18;025031850802;1;1;0;1;0;1;1;1
2003 18 5;19;025031850666;2;0;0;0;0;0;0;0
2003 18 5;20;025031850746;2;1;0;0;0;1;0;0
2003 18 5;21;025031850856;2;0;0;0;0;1;0;0
2003 18 5;22;025031830202;9;1;0;1;1;1;0;1
2003 18 5;23;025031850767;2;0;0;0;0;0;0;0

Dans cet exemple, on traite 4 contraintes de ratio prioritaires, 3 ratios non prioritaires, 8 véhicules sur la journée J-1 et 23 véhicules sur la journée J.

Journée J-1 : « *2003 18 3* »
Journée J : « *2003 18 5* »

Comme on le voit sur l'exemple, les journées J-1 et J ne sont pas nécessairement consécutives. Il s'agit de journées travaillées dans les usines, hors jours fériés et dimanches. Dans l'exemple, la date « 2003 18 4 » correspond au 1er mai.

Pour chaque véhicule, on donne d'abord les flags sur les contraintes de ratio prioritaires, puis sur les contraintes de ratio non prioritaires. Ces ratios sont dans le même ordre que dans le fichier des contraintes de ratio.

On liste d'abord les véhicules figés (journée J-1) avant de lister les véhicules à ordonnancer (journée J).

2.2 – Fournitures attendues des candidats

2.2.1-Dossier de réponse

- une fiche descriptive de l'équipe qui a travaillé sur le projet :
  - nom(s),
  - statut(s) (étudiant, chercheur, industriel...),
  - organisme(s) d'appartenance,
- présentation succincte (moins d'une page) de l'équipe et de ses travaux;
- un résumé étendu sur la ou les méthodes utilisées pour résoudre le problème posé avec les justifications et/ou les références nécessaires (4-6 pages). Il doit de plus comporter la liste des logiciels (commerciaux ou libres) utilisés ainsi que le langage de développement choisi.
- Un tableau excel de synthèse des résultats obtenus (nombre de violations de contraintes de ratio prioritaires, nombre de violations de contraintes de ratio non prioritaires, nombre de purges, note globale selon la règle établie ci-après) sur les diverses instances dans le temps imparti, **i.e. au bout de 600 secondes de temps CPU** (sur une machine comparable au PC décrit ci-après). Ce tableau excel aura le format suivant :

Nom instance	Nb violations contraintes de ratio prioritaires	Nb violations contraintes de ratio non prioritaires	Nb Purges	Note de l'instance

Pour les algorithmes non déterministes, les équipes donneront les meilleurs résultats obtenus en 10 exécutions, et la moyenne des résultats obtenus sur les 10 exécutions.

2.2.2 - Fichiers de solutions

Les fichiers de solutions associés aux différentes instances proposées doivent être fournies. Le fichier de solution pour une instance de problème doit avoir le format suivant : « Numéro séquence; PJI ». Le numéro de séquence commence à 1. Le nom du fichier de solutions sera le nom de l'instance associée.

Exemple de fichier solution :

```
 1;025031910825
 2;025031820293
 3;025031850497
 4;025031911085
 5;025031850299
 6;025031850082
 7;025031850783
 8;025031920415
 9;025032010270
10;025031911083
11;025031910060
12;025031970126
```

2.2.3 - Programmes

Pour le programme ayant permis de produire ces solutions, deux possibilités sont offertes :

- de préférence un fichier directement exécutable sur la machine cible.
- ou un ensemble de fichiers sources C ou C++, fournis avec une documentation pour la compilation et le fichier `makefile` associé, l'ensemble devant permettre de générer sans difficulté l'exécutable sur la machine cible.

2.2.4 - Organisation de l'arborescence des dossiers candidats

Pour un participant indicé `NN` dans l'ordre d'arrivée, l'arborescence sera la suivante :

- répertoire: `Candidate-NN/`
- sous-répertoires:

```
Candidate-NN/Team-description/
Candidate-NN/Method-description/
Candidate-NN/Result-synthesis/
Candidate-NN/Instances/
Candidate-NN/Solutions/
Candidate-NN/Results/
Candidate-NN/Program/
```

L'exécutable doit pouvoir être utilisé en ligne en tapant la commande suivante, à partir du répertoire `Candidate-NN/Program/` :

```
executable-name instance-name -t cpu-time
```

avec :

`instance-name` : nom du répertoire contenant les données de l'instance à traiter
`cpu-time` : **temps de calcul imparti en secondes, fixé à 600.**

Le programme devra aller chercher les données d'entrée de l'instance de nom `instance-name` dans le répertoire `Candidate-NN/Instances/instance-name` et écrire la solution obtenue au bout du temps `cpu-time` dans le répertoire `Candidate-NN/Solutions/`. Le fichier solution aura pour nom `instance-name`.

2.3 - Procédure d'évaluation et de classement

Il y a deux catégories :

- la catégorie **Senior** qui regroupe globalement tous les candidats,
- la catégorie **Junior** limitée aux projets dont les équipes sont majoritairement constituées d'étudiants éventuellement encadrés par leurs professeurs.

Remarque

Un projet Junior peut aussi gagner le prix de la catégorie Senior s'il s'avère que le projet est le meilleur de tous de ce Challenge. En revanche, un projet Senior ne peut pas gagner le prix de la catégorie Junior.

2.3.1 – Temps imparti et plateforme logicielle/matérielle

On allouera au calcul un temps maximal de **600 secondes** sur un **PC Pentium4/1.6 Ghz/win2000/1 Go RAM**. C'est le temps maximal acceptable pour les utilisateurs opérationnels.

Les programmes pourront tourner dans un environnement Win2000 ou Linux Redhat 7.3.

Les logiciels commerciaux nécessitant des licences sont interdits.

2.3.2 – Programme d'évaluation

Le programme d'évaluation de RENAULT consistera en un exécutable qui, sur une instance, appellera le programme fourni par un candidat, lui laissera **600 secondes**, ira récupérer la solution qui devra avoir été écrite par le programme dans le répertoire `Candidate-NN/Solutions/`, la vérifiera, l'évaluera et ira écrire les indicateurs associés dans le répertoire `Candidate-NN/Results/`.

Pour les programmes non-déterministes, 10 exécutions par instance sont lancées afin d'obtenir une moyenne.

2.3.3 - Notation sur une instance de problème

Nous décrivons ci-après la notation pour la 1ère phase de qualification. Cette notation pourra évoluer à la marge pour la 2ème phase et la phase finale.

On calculera une note pondérée sur les différents objectifs d'optimisation appliqués à l'instance de problème. Cela donnera les notes suivantes selon l'ordre des objectifs d'optimisation :

2.3.3.1 – Cas d'une instance avec « contraintes de ratio en priorité »

Ordre des objectifs d'optimisation :
(1) minimiser le nombre de violations des contraintes de ratio prioritaires
(2) minimiser le nombre de purges de peinture
(3) minimiser le nombre de violations des contraintes de ratio non prioritaires

note de l'instance = (10000*nb_violations_ratios_prioritaires) + (100*nb_purges) + (nb_violations ratios_non_prioritaires)

ou

Ordre des objectifs d'optimisation :
(1) minimiser le nombre de violations des contraintes de ratio prioritaires
(2) minimiser le nombre de violations des contraintes de ratio non prioritaires
(3) minimiser le nombre de purges de peinture

note de l'instance = (10000*nb_violations_ratios_prioritaires) + (100* nb_violations ratios_non_prioritaires) + (nb_purges)

ou

Ordre des objectifs d'optimisation :
(1) minimiser le nombre de violations des contraintes de ratio prioritaires
(2) minimiser le nombre de purges de peinture

note de l'instance = (10000*nb_violations_ratios_prioritaires) + (100*nb_purges)

2.3.3.2 – Cas d'une instance avec « rafales de peinture en priorité »

Ordre des objectifs d'optimisation :
(1) minimiser le nombre de purges de peinture
(2) minimiser le nombre de violations des contraintes de ratio prioritaires
(3) minimiser le nombre de violations des contraintes de ratio non prioritaires

note de l'instance = (10000*nb_purges) + (100*nb_violations_ratios_prioritaires) + (nb_violations_ratios_non_prioritaires)

ou

Ordre des objectifs d'optimisation :
(1) minimiser le nombre de purges de peinture
(2) minimiser le nombre de violations des contraintes de ratio prioritaires

note de l'instance = (10000*nb_purges) + (100*nb_violations_ratios_prioritaires)

2.3.4 - Notation globale sur l'ensemble des instances de problème d'une base

Pour chaque base, on regroupe les instances en 3 séries :

- instances avec « contraintes de ratio en priorité » et contraintes de ratio prioritaires **faciles**
- instances avec « contraintes de ratio en priorité » et contraintes de ratio prioritaires **difficiles**
- instances avec « rafales de peinture en priorité »

Pour chaque série et chaque équipe candidate, on calcule la **moyenne** des notes obtenues par l'équipe sur les différentes instances de la série (la note de chaque instance est calculée selon la notation définie en 2.3.3).

Sur l'ensemble des 3 séries, on calculera ensuite la note globale suivante, pour classer les équipes candidates :

$$\sum_{serie} \left[\frac{(note_candidat(serie) - moins_bonne_note(serie))}{(meilleure_note(serie) - moins_bonne_note(serie))} \right]$$

La « moins bonne note » et la « meilleure note » sont la moins bonne note sur l'ensemble des équipes candidates, et la meilleure note sur l'ensemble des équipes candidates, pour une série donnée d'instances.

Cette note est l'unique critère de classement des équipes. Cette formule permet de ramener les notes moyennes sur les 3 séries dans le même ordre de grandeur.

Sur les résultats de la base A, au plus **10 candidats** seront retenus pour participer à la phase finale.

Les finalistes seront départagés sur les instances de la base X. La base B ne sert qu'aux réglages des programmes des finalistes.

2.4 – Planning du challenge

15/07/2003 : début de la phase de qualification
mise en place du site WEB, avec le sujet complet et **une instance de problème par série** (soit 3 instances au total)

01/10/2003 : diffusion de toutes les instances de problèmes de la base A

01/12/2003 : fin du dépôt des candidatures,
pour vous inscrire, veuillez envoyez vos affiliations complètes par email à Van-Dat CUNG

29/02/2004 : **fin de la phase de qualification**
date limite pour la fourniture des résultats obtenus sur les instances de la base A

17/05/2004 : **début de la phase finale**

- annonce des résultats à l'issue de la 1ère phase, sélection des finalistes qui seront invités à présenter leurs travaux au congrès ROADEF'2005
- diffusion des instances de problèmes de la base B

Cette base B permettra aux équipes finalistes, qui le souhaitent, de régler leurs programmes

03/10/2004 : date limite pour fournir une version améliorée des programmes
RENAULT testera les programmes fournis sur les instances de la base X inconnue des candidats

Congrès ROADEF'2005 : **fin de la phase finale**
Annonce des résultats finaux sur la base X.
Présentation par les finalistes de leurs travaux.
Le gagnant dans chacune des catégories se verra attribuer un prix par l'intermédiaire de la ROADEF.