

Simplification de données de vol pour un stockage optimal et une visualisation accélérée

Ibrahim Chahid*,**
Sofian Maabout**

Loïc Martin**
Mohamed Mosbah**

*2MoRO, Bidart France
ibrahim.chahid@2moro.fr,

**LaBRI, IPB, Université Bordeaux1, France
{loic.martin, maabout, mosbah}@labri.fr

Résumé. Le projet RECORDS (collaboration entre industriels et université) a pour objectif de développer une infrastructure de service sécurisée pour assurer le suivi et l'analyse des conditions d'utilisation d'aéronefs. Chaque aéronef est muni de capteurs. Au cours de chaque mission (vol) les données mesurées sont enregistrées localement. Ces dernières sont par la suite transférées dans une base de données centralisée à des fins d'analyse. Le problème rencontré est la grande quantité de données ainsi enregistrées, ce qui en rend l'exploitation difficile. Dans cet article, nous proposons des techniques de compression et de simplification de données avec un taux de perte contrôlé. Nos expérimentations montrent des gains drastiques en volumétrie avec de très faibles pertes d'informations. Ceci représente une première étape avant d'appliquer des techniques d'extraction de connaissances.

1 Introduction

Le projet RECORDS¹ a pour objectif de développer une infrastructure de service sécurisée pour assurer le suivi et l'analyse des conditions d'utilisation d'aéronefs.

Chaque aéronef est muni de capteurs. Au cours de chaque mission (vol) les données mesurées sont enregistrées localement par l'intermédiaire d'un système embarqué. Ces dernières sont par la suite transférées dans une base de données centralisée à des fins d'analyse. Cette analyse permet de veiller au respect des conditions d'utilisation imposées par la réglementation et les consignes constructeur et aide à l'optimisation de la gestion d'une flotte.

Le problème rencontré est la grande quantité de données ainsi enregistrées. En effet, la position d'un véhicule est capturée à une grande fréquence. ceci sans compter les autres données de vol. Cette volumétrie est très importante sur le long terme vis-à-vis de la base de données stockant l'ensemble des enregistrements en vue d'analyses.

L'accès à la base de données et à l'ensemble des analyses se fait par un portail Web sécurisé. Les contraintes matérielles actuelles ne permettent pas de faire transiter rapidement de

¹projet mené par un consortium de partenaires industriels avec le LaBRI : 2MORO, Nexter Systems, Socata, Turbomeca, TDM Ingenierie, FEDD et Thales Avionics

si grandes quantités de données via le réseau internet. Or pour que l’outil de visualisation soit utilisable de manière régulière il faut que le temps d’affichage des valeurs, et leur géolocalisation sur les trajectoires, soient les plus réduits possible. Nous espérons avoir des temps de l’ordre de la minute.

Dans cet article, nous proposons des techniques de compression et de simplification de données avec un taux de perte contrôlé. Nos expérimentations montrent des gains importants en volumétrie avec de très faibles pertes d’informations. Ceci représente une première étape avant d’appliquer des techniques d’extraction de connaissances.

La prochaine section introduit le principe de nos méthodes de simplification et les techniques de reconstruction permettant de retrouver des données proches des originaux. Dans la section 3 nous détaillons une méthode rapide pour simplifier l’ensemble des données à travers l’exemple complexe de la trajectoire. Cette première méthode ne donnant pas toujours des résultats satisfaisants nous proposons, dans la section 4, une autre approche plus précise.

2 Les principes de la simplification et de l’interpolation des données

Afin de répondre au mieux aux contraintes de volumétries et de visualisation, nous proposons une simplification des données couplée à une méthode d’interpolation efficace.

Pour chaque capteur c_i on a un ensemble \mathcal{P}_i de paires (t_{ij}, m_{ij}) où m_{ij} est une mesure captée par c_i et t_{ij} désigne le moment de la prise de la mesure.

Ces paires décrivent une courbe qu’on peut approximer par des fonctions d’interpolation. Ainsi, toute la courbe (ensemble de paires) est résumée par une fonction. Cependant, nous n’avons pas adopté cette solution car la recherche de cette fonction (ses coefficients) est complexe.

L’utilisation des NURBS², décrites dans Piegl et Tiller (1995) et permettant de dessiner la courbe grâce à des points de contrôle extérieurs, n’a pas été retenue car même si l’affichage est très rapide, trouver la position des points de contrôle de manière automatique est coûteux.

Le principe de la technique que nous utilisons pour réduire cet ensemble de points peut-être décrit comme suit : nous sélectionnons un sous-ensemble de \mathcal{P}_i qu’on considère comme étant “les points essentiels”. La restitution de la courbe se fait grâce à l’application de fonctions d’interpolations pour chaque paire de points essentiels successifs.

Cette méthode d’interpolation particulièrement efficace et proche d’un tracé lissé à la main est appelée la Spline cubique d’Hermite. Elle est expliquée dans Knott (2000).

Cette méthode permet de décrire une courbe entre deux points grâce à quatre fonctions de degré 3 qui toutes utilisent les positions initiale m_0 et finale m_1 , les dérivées au départ V_0^m et à l’arrivée V_1^m . Les fonctions d’Hermite sont uniques et déjà définies. Les dérivées sont approximées par le calcul dit de la “différence limitée”.

Les splines d’Hermite ne permettent de décrire qu’une seule dimension en fonction du temps. Néanmoins, la trajectoire (qui est en 3 dimensions) est facile à reconstituer en combinant les reconstructions issues de chacune des dimensions spatiales.

Nous allons détailler la simplification de la trajectoire qui est la plus complexe car il faut prendre en compte les 4 dimensions spatio-temporelles à la fois.

²Non-Uniform Rational Basis Splines

Soit $\mathcal{P}\{p_i = (t_i, L_i, l_i, a_i, c_i, V_i^L, V_i^l, V_i^a, D_i, \Delta_i Rec)\}$, L'ensemble des points d'une trajectoire caractérisés par leurs positions (L_i, l_i, a_i) , c_i qui représente le cap, (V^L, V^l, V^a) qui représente les vitesses respectives à L, l, a , D_i une approximation de la distance déjà parcourue depuis le début du vol et enfin $\Delta_i Rec$ la tolérance associée à ce point. Nous notons \mathcal{PE} le sous-ensemble de \mathcal{P} qui contient les "points essentiels" de \mathcal{P} .

3 Simplification de la trajectoire par l'étude comportementale

Notre première approche s'intéresse particulièrement à trois paramètres associés à la trajectoire : le cap du véhicule, son altitude et sa position (Longitude, Latitude).

D'autres paramètres, comme l'*attitude* de l'appareil en vol : assiette, roulis, lacet, ne sont actuellement pas pris en compte. Cependant, les modifications à apporter pour intégrer ces paramètres à l'algorithme sont minimales et permettront d'ailleurs de gagner en précision.

3.1 Sélection des points essentiels

Nous détaillons l'algorithme de sélection des "points essentiels". Ce dernier utilise deux curseurs qui parcourent l'ensemble de la trajectoire : le premier, nommé *position référence* ou *Ref*, sert à pointer la dernière position représentative de la trajectoire trouvée et l'autre, nommé *position à comparer* ou *Comp*, est utilisé pour avoir accès aux informations du point que l'on souhaite tester.

L'initialisation. Les valeurs seuils : ΔCap , $\Delta Altitude$ et $\Delta Distance$ doivent être choisies par l'utilisateur de façon à permettre le meilleur compromis entre gain volumétrique et précision. En effet, des valeurs Δ trop élevées permettent un gain volumétrique important mais impliquent une perte relativement conséquente en précision, et inversement.

Lorsque les seuils sont paramétrés, le premier point de la trajectoire est inclus dans les positions essentielles de la trajectoire et est pointé par le curseur *Ref*, le curseur *Comp* est associé à la position suivante.

Le parcours de la trajectoire. À chaque itération, nous comparons les paramètres des deux curseurs. Si les différences de cap, d'altitude ou de distance parcourue dépassent l'un des seuils fixés alors la *position à comparer* est sélectionnée comme étant représentative de la trajectoire du véhicule et l'on place *Ref* sur cette position et *Comp* sur la suivante. Sinon le point n'est pas retenue et seul le curseur *Comp* est déplacé sur la position suivante afin de recommencer la comparaison *Ref* et *Comp*.

L'algorithme se termine lorsque l'on atteint la dernière position enregistrée, qui est automatiquement retenue comme représentative.

Bien que très efficace (en temps de calcul), les points essentiels retournés par cet algorithme ne permettent pas toujours d'avoir une trajectoire reconstituée qui reflète fidèlement la trajectoire initiale. Pour cette raison, nous proposons un autre algorithme de sélection de ces points.

4 Simplification de la trajectoire par comparaison de la reconstruction avec l'originale

Le principe du second algorithme est le suivant : Chaque point p de la trajectoire initiale est comparé à son prédécesseur (dernier point avant p , selon l'ordre chronologique, à ne pas avoir été supprimé), s'ils sont *similaires* alors p est ajouté à \mathcal{S} (l'ensemble des points supprimés). La similarité est basée sur la vitesse en un point et est détaillée dans la section 4.1. Puisque cette suppression n'est basée que sur la vitesse, nous devons vérifier que les points de \mathcal{P} non supprimés permettent bien de retrouver les positions de ceux de \mathcal{S} avec un écart inférieur à $\Delta_{max}Rec$. Les éléments de \mathcal{S} qu'on n'arrive pas à approximer correctement sont considérés comme essentiels et sont transférés dans \mathcal{PE} . Si à la fin de ces deux étapes il reste des points de \mathcal{P} qui ne sont ni dans \mathcal{S} ni dans \mathcal{PE} alors on recommence avec un seuil de similarité plus élevé.

L'utilisateur définit l'écart maximum (entre le relevé et la reconstruction) qu'il autorise, noté $\Delta_{max}Rec$.

Afin de ne pas perdre des informations utiles pour l'utilisateur, comme d'éventuels trous d'air ou encore les phases de décollage/atterrissage, nous modulons l'écart autorisé en fonction de la vitesse instantanée horizontale de l'avion. L'écart autorisé en un point k est noté $\Delta_k Rec$.

4.1 Suppression des points non essentiels

Pour l'algorithme, nous appelons points successifs non pas deux points qui se suivent dans \mathcal{P} mais deux points successifs de $\mathcal{P} \setminus \mathcal{S}$. C'est à dire que p_i et p_j ($j > i$) sont des points successifs ssi $\{p_i, p_j \in \mathcal{P} \setminus \mathcal{S} \ \& \ \forall k \mid i < k < j; p_k \in \mathcal{S}\}$

Pour comparer deux points p_i et p_j , ($j > i$), on vérifie si $\|\vec{V}_i - \vec{V}_j\| < \epsilon$, avec $\vec{V}_k = (V_k^L, V_k^l, V_k^a)$ et ϵ un seuil. Dans ce cas, p_j est ajouté à \mathcal{S} car les deux points sont similaires.

Le calcul du seuil ϵ est totalement indépendant du choix de l'écart $\Delta_{max}Rec$.

Pour supprimer l'ensemble des points potentiellement non utiles à la reconstruction, on parcourt l'ensemble des points de $\mathcal{P} \setminus \{\mathcal{S} \cup \mathcal{PE}\}$ que l'on compare au point précédent.

Une fois arrivé à la fin de l'ensemble \mathcal{P} , de nombreux points ont été ajoutés à \mathcal{S} . Afin de s'assurer que la nouvelle reconstruction ne s'éloigne pas trop de la véritable trajectoire on fait une vérification de la reconstruction obtenue comme expliqué ci-après.

4.2 Vérification de la nouvelle trajectoire

Le principe est de chercher parmi les points de \mathcal{S} quels sont ceux dont la reconstruction s'éloigne trop de leur position réelle.

Pour la vérification, on utilise la méthode d'interpolation cubique de Hermite avec seulement les points de $\mathcal{P} \setminus \mathcal{S}$ comme points de contrôle.

Pour chaque point p_k de \mathcal{S} , on calcule, grâce à $\mathcal{P} \setminus \mathcal{S}$, sa nouvelle position $p_k^c = \{L_k^c, l_k^c, a_k^c\}$ en récupérant le t_k associé à ce point et en utilisant le calcul des Splines d'Hermite.

On vérifie si la distance entre la position calculée $\{L_k^c, l_k^c, a_k^c\}$ et la position réelle $\{L_k, l_k, a_k\}$ du point est supérieure à l'écart autorisé $\Delta_k Rec$. Dans ce cas, le point p_k est considéré comme essentiel et il est supprimé de \mathcal{S} pour être ajouté à \mathcal{PE} .

Une fois l'ensemble des points de \mathcal{S} vérifiés, si $\mathcal{S} \cup \mathcal{PE} \neq \mathcal{P}$ alors on augmente le seuil ϵ (rendant ainsi plus de points similaires), et on recommence l'étape précédente de suppression.

5 Expérimentation

Les deux méthodes n'utilisent pas les mêmes paramètres pour la simplification de la trajectoire et donc ne sélectionnent pas les mêmes points essentiels.

La première méthode : Le gros avantage de cette méthode est sa vitesse d'exécution ; il lui suffit d'un passage sur l'ensemble des points \mathcal{P} pour sélectionner tous les points essentiels. Par contre, l'utilisateur ne peut pas savoir la précision de la reconstruction a priori. Aussi, cette méthode peut s'avérer particulièrement pas précise quand notamment on est en présence de trajectoires saccadées. Il peut aussi avoir un mauvais positionnement du point sur la trajectoire (trop en avance ou en retard) qui ne se voit pas lorsque l'on regarde l'ensemble de la trajectoire reconstruite.

La seconde méthode : Cette méthode est plus longue, il faut une vingtaine de passages sur l'ensemble des points de \mathcal{P} pour trouver tous les points essentiels. La reconstruction de la trajectoire est plus fidèle à l'originale. Par contre, visuellement la nouvelle trajectoire peut s'écarter un peu plus qu'avec la première méthode surtout dans les virages. En effet, alors que la première méthode garde de nombreux points dans les virages (à cause du changement de cap), cette méthode supprime tous les points qui peuvent être reconstruits avec la précision requise même si cela autorise de "couper" dans les virages.

6 Conclusion

Nous avons présenté dans cet article deux techniques permettant de résumer les trajectoires d'objets mobiles (aéronefs). Ces techniques sont utilisables pour n'importe quelles autres mesures obtenues par des capteurs. Nous avons comparé la performance des deux méthodes en considérant les paramètres de temps d'exécution et la qualité de restitution. Comme l'on pouvait s'y attendre, la méthode la plus lente est celle qui engendre le moins de perte.

Dans nos travaux futurs, une des directions de recherche que nous comptons explorer consiste à confronter nos techniques de *compression* au domaine du SOLAP (Spatial On Line Analytical Processing) Bimonte et al. (2006); Bejaoui et al. (2008); Sboui et al. (2008). En effet, dans ce contexte, il est souvent question d'opérations d'agrégation et de *Roll Up, Drill Down* selon des dimensions hiérarchisées en termes de régions. Nous pensons que nos techniques permettront, peut être en les adaptant, d'optimiser les opérations classiques que l'on retrouve dans ce contexte.

Un autre axe que l'on compte poursuivre est celui des bases de données d'objets mobiles Cheng et al. (2004); Pfoser et al. (2000); Abdessalem et al. (2006). En effet, contrairement à l'application que nous avons décrite dans ce papier, la particularité de ces bases est que les données de géolocalisation arrivent sous forme de flux instantanées. Il serait donc intéressant de disposer de techniques de *compression* pouvant travailler à la volée.

Remerciements Ce travail a été réalisé dans le cadre du projet RECORDS financé par la Direction Générale des Entreprises (DGE) du ministère de l'industrie.

Références

- Abdessalem, T., L. Decreusefond, et J. Moreira (2006). Evaluation of probabilistic queries in moving objects databases. In *Fifth ACM International Workshop on Data Engineering for Wireless and Mobile Access, Mobide*.
- Bejaoui, L., F. Pinet, M. Schneider, et Y. Bédard (2008). An adverbial approach for the formal specification of topological constraints involving regions with broad boundaries. In *proceedings of ER'08 conference*, pp. 383–396.
- Bimonte, S., A. Tchounikine, et M. Miquel (2006). GeoCube, a Multidimensional Model and Navigation Operators Handling Complex Measures : Application in Spatial OLAP. In T. Y. Erich Neuhold (Ed.), *Fourth Biennial International Conference on Advances in Information Systems*, pp. 100–109. Springer-Verlag.
- Cheng, R., D. V. Kalashnikov, et S. Prabhakar (2004). Querying imprecise data in moving object environments. *IEEE Trans. Knowl. Data Eng.* 16(9), 1112–1127.
English
- Knott, G. (2000). *Interpolating Cubic Spline*. Birkhäuser.
- Pfoser, D., C. S. Jensen, et Y. Theodoridis (2000). Novel approaches in query processing for moving object trajectories. In *VLDB*.
- Piegl, L. et W. Tiller (1995). *The NURBS book*. London, UK : Springer-Verlag.
- Sboui, T., M. Salehi, et Y. Bédard (2008). Catégorisation des problèmes d'intégration des modèles des cubes de données spatiales. In *Atelier-Fouille de données complexes- EGC'08*.

Summary

The RECORDS project (collaboration between industry and university) aims to develop an infrastructure and a secure service for monitoring and analyzing use conditions of aircrafts. Each aircraft is equipped with sensors. During each mission (flight) data measurements are recorded locally. These are subsequently transferred in a centralized database for analysis. The problem we encountered before analyzing these recorded data is their huge amount, which makes their treatment difficult. In this paper, we propose compression techniques and simplification of data with a controlled loss rate. Our experiments show dramatic gains in volume along a very low loss of information. This represents a necessary first step before applying knowledge extraction.