

Capítulo 6

Conclusiones

Se ha puesto de manifiesto la existencia de una creciente necesidad de planificar y gestionar de manera óptima los datos obtenidos a través de las flotas de EOSs, debido al crecimiento exponencial de las necesidades de datos de la Tierra, exigidos por un número de aplicaciones cada vez mayor. Actualmente la tendencia es una planificación multisatélite, lo cual conlleva la ventaja de llevar a cabo la adquisición de datos de una manera más rápida. No obstante, la complejidad del diseño de estas misiones requiere nuevos y mejores algoritmos capaces de manejar la complejidad de los parámetros que definen la misión (particularmente las relacionadas con la geometría). Los requisitos son la capacidad de trabajar en tiempo real y la obtención de una elección óptima de satélites (o al menos razonablemente cercana al óptimo).

Se ha realizado el modelado matemático del problema teniendo en cuenta diversos casos con distinta complejidad (inclusión de modos de sensor o no). Se ha planteado el problema como uno de programación lineal binaria, y se han propuesto algoritmos heurísticos para la resolución. En el curso de estas formulaciones se hace patente la necesidad de un programa de geometría computacional que lleve a cabo las distintas operaciones con polígonos que se requieren para ambas formulaciones. La herramienta escogida ha sido PolygonClip, debido a su sencilla integración con Matlab y a su nulo coste. No obstante, se ha detectado cierta problemática asociada a la herramienta, que hace necesarias labores de depuración y aumenta los tiempos de computación. Otras herramientas tal vez no conlleven estos problemas, pero su uso está fuera del alcance de este proyecto.

Para la resolución del problema una vez formulado se ha utilizado la función de Matlab *bintprog*. En algunos casos donde esta función no era capaz de resolver el problema se ha utilizado una herramienta gratuita de código abierto llamada *lpsolve*, con la cual dichos problemas quedaban resueltos en poco tiempo. El uso de esta herramienta, al ser

gratuita (al contrario que Matlab) y ofrecer mejores tiempos, podría mejorar la resolución del problema.

El uso de algoritmos exactos se reduce sencillamente al cálculo de la matriz Q . Esta simplifica de una manera muy efectiva toda la información relativa a las adquisiciones y subregiones generadas, permitiendo una formulación matemática sencilla para resolver el problema. Es por tanto la base del algoritmo y dada su importancia se debe desarrollar una manera eficaz de obtenerla. La herramienta de geometría computacional tiene un peso enorme, tanto en tiempo como en precisión. Si bien PolygonClip tiene ciertos fallos en ambos campos, las ventajas antes expuestas siguen convirtiéndolo en la mejor opción para los objetivos de este proyecto.

Para el uso de algoritmos heurísticos la influencia de la geometría no es tanta. Al ser más sencillos los tiempos de computación no son tan altos, pero presentan la desventaja de no garantizar una solución óptima. No obstante, si se observa que las soluciones están razonablemente cerca de las obtenidas con algoritmos exactos, se puede concluir que lo estarán en el resto de casos. Esto en conjunto al ahorro en tiempo y memoria los convierten en los mejores candidatos a solucionar el problema de planificación de EOSs.

Con respecto al parámetro k utilizado en los algoritmos heurísticos, a la vista queda que es mejor el uso de valores pequeños. Al aumentar mucho no se consigue una mejora de la solución (incluso empeora en ocasiones), y el tiempo de computación va aumentando paulatinamente. Esto es lógico ya que aumentar k conlleva proponer candidatos a la solución que ofrecen una mejora de la función objetivo cada vez menor, con lo que adquisiciones que no aportan demasiado a la resolución del problema pasan a formar parte de la solución definitiva de éste.

También se ha extraído como conclusión que las variables que más influyen en el tiempo de computación son el tamaño y número de adquisiciones, que a su vez dependen del tamaño de la región de interés, del número de satélites incluidos en la base de datos, del número de modos que tiene cada satélites y del ángulo del sensor del satélite.

A la vista queda además la inviabilidad de obtener una solución exacta con los métodos planteados en este proyecto. Los tiempos de obtención de la solución aumentan rápidamente con el número de satélites, los modos y el área de la región de interés. Para regiones pequeñas el incluir los modos lleva al límite estos algoritmos, sobre todo si se quiere una solución en tiempo real. Para regiones mayores incluso sin tener modos en los satélites los tiempos son prohibitivos. Al añadir los modos tanto la memoria como el tiempo crecen enormemente.

Por todo esto los algoritmos heurísticos se destacan como los claros candidatos para resolver el problema. La única desventaja que presentan, no garantizar una solución óp-

tima, queda ampliamente subsanada por los bajos consumos en tiempo y memoria. De hecho la desventaja no es tal si se tiene en cuenta que las soluciones obtenidas están razonablemente cerca de las obtenidas mediante la resolución exacta del problema.

Trabajo futuro

El objetivo de este proyecto es la resolución del problema de adquisición de imágenes mediante EOSs aplicado a satélites reales. Si bien el trabajo realizado en este campo ha sido exhaustivo, quedan aún así opciones de ampliación. Ha quedado fuera del proyecto, aunque sería de interés, el estudio de otras funciones objetivo, tales como el minimizar el tiempo de obtención de la imagen, y restricciones más desfavorables en cuanto a los satélites, como las debidas al *dutycycle*.

Asimismo, como se comenta en la sección 3.2, se ha considerado la Tierra para el cálculo de latitud y longitud como una esfera perfecta. Esto no es cierto (de hecho su no esfericidad se ha tenido en cuenta en el propagador de órbitas), y se podrían considerar geometrías más complejas tales como el Elipsoide Internacional WGS84 de la figura 3.4 del apartado citado anteriormente. No obstante, las variaciones son pequeñas.

En cuanto al modelado de la geometría, el cálculo de áreas no es exacto. Para los datos de áreas utilizados en el algoritmo heurístico éstas se calculan como si las adquisiciones y los polígonos generados fuesen polígonos planos haciendo corresponder las abscisas con la longitud y las ordenadas con la latitud, en proyección cilíndrica equidistante. Para regiones que no sean cercanas a los polos la aproximación es lo suficientemente buena, pero obviamente a mayor distancia del ecuador mayor es la desviación. Se podría implantar un módulo de geometría computacional 3D en lugar de 2D para evitar estas inexactitudes, que dada su complejidad queda fuera del alcance de este proyecto.