UDC 528.852.1

COMPUTATIONAL COMPLEXITY OF ALGORITHMS FOR FORMATION OF RADAR IMAGES

T. DROVOSEKOVA, R. BOHUSH
Polotsk State University, Belarus

In this paper the basic algorithms for spaceborne Synthetic Aperture Radar data-focusing are considered. The comparative analysis of computational complexity for them is presented.

Introduction. The process of forming a radar image is called focusing of a radio hologram and is performed at the stage of primary processing of radar data. It includes irradiation of the Earth's surface with a probing signal, reception and processing of echo signals (responses) reflected from points on the Earth's surface, and the actual formation of radar images.

Currently, to get high spatial resolution of the obtained radar images, radars with a synthetic antenna aperture (SAR) are used. When moving in orbit, the SAR antenna emits short frequency or phase modulated pulses with a given repetition rate. The synthesis of the aperture is the result of successive reception of the responses of the probing signals by the antenna at a different position relative to the source of these responses. For each radiation of the probing signal, one line of the radio hologram is recorded. The number of radio hologram lines depends on the frequency of the probing signals and the flight time of the radar.

Types of computational complexity of algorithms. By computational complexity we understand the function of work volume dependence that is performed by some algorithm on the size of the input data. The volume of work is usually measured by abstract concepts of time and space, called computing resources. The theory of computational complexity arose from the need to compare the performance of algorithms, clearly describe their behavior (runtime and amount of required memory) depending on the size of the input data. The time complexity of an algorithm is a function of the size of the input data, equal to the maximum number of elementary operations performed by the algorithm to solve an instance of a task of a specified size. By analogy with time complexity, the spatial complexity of the algorithm is determined, only here they show not only the number of elementary operations, but the amount of memory used. Despite the fact that the function of the time complexity of the algorithm in some cases can be determined exactly, in most cases it is meaningless to look for its exact value, because, firstly, the exact value of the time complexity depends on the definition of elementary operations, and secondly, with increasing the size of the input data, the contribution of constant factors and lower order terms appearing in the expression for the exact time of operation becomes extremely insignificant.

Consideration of large input data and estimation of the operating time of order growth, the algorithm leads to the concept of asymptotic complexity of the algorithm. Moreover, an algorithm with less asymptotic complexity is more efficient for all input data, with the possible exception of, possibly, small data. Currently, the asymptotic complexity of most mathematical operations, functions, and transformations is calculated.

Analysis of the computational complexity of focusing algorithms. For primary processing of PCA data, the following algorithms are used: Range Doppler (RD), Chirp Scaling Algorithm (CSA), Range Migration Algorithm (RMA), as well as their modifications: Range Doppler Algorithm (RDA), Extended Nonlinear Chirp Scaling Algorithm (ECS).

The essence of the RD and RDA algorithms is as follows: using the fast Fourier transform (FFT) method, the spectrum of the input signal is obtained, the resulting spectrum is multiplied by the spectrum of the reference function (range compression), and the inverse Fourier transform (IFFT)) is performed. For azimuth compression, matched filtering in the frequency domain in azimuth is used. The change in the Doppler frequency depends on the range, therefore, a separate filter is used for each column of the RGG. To compensate for range migration, an array of range-compressed data is converted into range / Doppler frequency coordinates and each column of the data array is added with the corresponding frequency offset (RCMC – Range Cell Migration Correction) [1].

The CSA and ECS algorithms are designed based on alignment of curvature (range migration) in the two-dimensional frequency domain so as to have congruent range migration paths. This is achieved by the successive use of FFT in azimuth, multiplying the obtained spectrum by the LFM support (LFM scaling), FFT in range, multiplying by the phase function (correction of average range migration) and subsequent IFFs in range with phase correction distorted during LFM scaling and IFFT in azimuth [2].

The RMA algorithm involves the initial conversion of signal data into a two-dimensional frequency domain. The second focusing step is multiplying by the reference function calculated for the selected range. This operation can be considered as "volume compression". The third stage of focusing is Stolt interpolation (transformation), which completes the focusing of targets outside the control range. Finally, two-dimensional IFFT is performed to convert the data back to the time domain, that is, to the image region [3].

To compare the computational complexity of the considered algorithms, the amount of input data should be the same for each algorithm and have a size of Na × Nr (azimuth × range). As a result of focusing, the obtained radar image must also have proportional characteristics. In this case, the temporal complexity of the algorithm will be the characteristic that determines the best algorithm from the point of view of computational complexity.

The complexity of the focusing algorithms at each stage is studied in terms of the number of floating point operations (FLOP). Each FLOP can be either real multiplication or real addition. Complex multiplication of floating point numbers requires 6NrNa operations. FFT or IFFT can be calculated as 5NrNalog2Nr/a operations (depending on range or azimuth conversion). Interpolations of various types (sinc, Lagrange, Stolt) have complexity 2 (Mken - 1) NaNr, where Mken is the length of the interpolation core [4].

Table shows the results of determining the computational complexity of the algorithms.

Table. – Results of determining the computational complexity of algorithms

<table>
<tr><th rowspan="2">The main steps of the algorithm</th><th colspan="5">Computational complexity, FLOP</th></tr>
<tr><th>RD</th><th>RDA</th><th>CSA</th><th>ECS</th><th>RMA</th></tr>
<tr><td>Range FFT</td><td>5NaNrlog$_2$Nr</td><td>5NaNrlog$_2$Nr</td><td>5NaNrlog$_2$Nr</td><td>5NaNrlog$_2$Nr</td><td>–</td></tr>
<tr><td>Range IFFT</td><td>5NaNrlog$_2$Nr</td><td>5NaNrlog$_2$Nr</td><td>5NaNrlog$_2$Nr</td><td>5NaNrlog$_2$Nr</td><td>–</td></tr>
<tr><td>Azimuth FFT</td><td>5NaNrlog$_2$Na</td><td>5NaNrlog$_2$Na</td><td>5NaNrlog$_2$Na</td><td>5NaNrlog$_2$Na</td><td>–</td></tr>
<tr><td>Azimuth IFFT</td><td>5NaNrlog$_2$Na</td><td>5NaNrlog$_2$Na</td><td>5NaNrlog$_2$Na</td><td>5NaNrlog$_2$Na</td><td>–</td></tr>
<tr><td>Two-dimensional FFT</td><td>–</td><td>–</td><td>–</td><td>–</td><td>5NaNrlog$_2$Na+
+5NaNrlog$_2$Nr</td></tr>
<tr><td>Two-dimensional IFFT</td><td>–</td><td>–</td><td>–</td><td>–</td><td>5NaNrlog$_2$Na+
+5NaNrlog$_2$Nr</td></tr>
<tr><td>Azimuth Average Compression</td><td>–</td><td>–</td><td>–</td><td>–</td><td rowspan="3">24NaNr+
+2(Mken_l – 1)*
*NaNr</td></tr>
<tr><td>Range Compression</td><td>6NaNr</td><td>6NaNr</td><td rowspan="3">6NaNr</td><td rowspan="3">6NaNr</td></tr>
<tr><td>Medium Migration Correction</td><td>–</td><td>–</td></tr>
<tr><td>Secondary compression</td><td>–</td><td>6NaNr</td><td>–</td></tr>
<tr><td>Range Migration Correction</td><td>18NaNr+
+2(Mken_l – 1)*
*NaNr</td><td>18NaNr+
+2(Mken_l – 1)*
*NaNr</td><td>–</td><td>–</td><td>–</td></tr>
<tr><td>Resolving Range Migration Differences</td><td>–</td><td>–</td><td>6NaNr</td><td>6NaNr</td><td>(Mken_s – 1)*
*NaNr</td></tr>
<tr><td>Azimuth compression</td><td>6NaNr</td><td>6NaNr</td><td rowspan="2">6NaNr</td><td>6NaNr</td><td>–</td></tr>
<tr><td>Phase correction</td><td>–</td><td>–</td><td rowspan="2">6NaNr</td><td>–</td></tr>
<tr><td>Elimination of differences in the rate of change of the Doppler frequency</td><td>–</td><td>–</td><td>–</td><td>–</td></tr>
<tr><td>Total computational complexity</td><td>10NaNrlog$_2$Na+
+10NaNrlog$_2$Nr+
30NaNr+
+2(Mken_l – 1)*
*NaNr</td><td>10NaNrlog$_2$Na+
+10NaNrlog$_2$Nr+
+36NaNr+
2(Mken_l – 1)*
*NaNr</td><td>10NaNrlog$_2$N
a+
+10NaNrlog$_2$
Nr+
+18NaNr</td><td>10NaNrlog$_2$N
a+
+10NaNrlog$_2$
Nr+
+24NaNr</td><td>10NaNrlog$_2$Na+
+10NaNrlog$_2$Nr+
+24NaNr+
+2(Mken_l – 1)*
NaNr+2(Mken_s –
1)*
*NaNr</td></tr>
</table>

As can be seen from the table, the CSA algorithm has the lowest computational complexity, while it provides high quality primary processing of SAR data in conditions when the deviations from the strictly side view (at which the Doppler centroid is zero) are small. If these conditions are not satisfied, then deviations of the law of change of the Doppler frequency from the linear one and the dependence of the Doppler centroid on the distance arise. In the CSA and ECS algorithms, these facts are not taken into account, which leads to a decrease in the quality of the synthesized radar images.

The RD and RDA algorithms have the greatest flexibility and allow to obtain high-quality radar images even with sufficiently large deviations from strictly lateral viewing. But at the same time they require more floating-point operations than CSA and ECS algorithms.

The RMA algorithm is accurate even if the aperture is very wide or there is a significant deviation of the radar from a strictly side view, however, Stolt interpolation requires a significant amount of mathematical operations. The RMA algorithm is not used to focus radar data from space SAR, because it is based on the Stolt transform.

REFERENCES

1. Po-Chih Chen, Jean-Fu Kiang. An Improved Range-Doppler Algorithm for SAR Imaging at High Squint Angles. Progress In Electromagnetics Research M, Vol. 53, 41–52, 2017
2. R.K. Raney, H. Runge, R. Bamler, I.G. Cumming, F.H. Wong. Precision SAR Processing Using Chirp Scaling. IEEE Transactions on Geoscience and Remote Sensing, 32(4):786-799, July 1994
3. I.G. Cumming, Y.L. Neo, F.H. Wong. Interpretations of the Omega-K Algorithm and Comparisons with other Algorithms. In Proceedings of the IEEE International Geoscience and Remote Sensing Symposium, IGARSS'03, Toulouse, July 21-25, 2003
4. Y. Yuan, S. Chen, S. Zhang, H. Zhao. A chirp scaling algorithm for forward-looking linear-array SAR with constant acceleration. IEEE Geosci. Remote Sens. Lett. 2018, 15, 88–91.