

Pythonでレッスン…自己位置の割り出しや衛星軌道の算出

廣川 類

現在では各国が多くの測位衛星を運用しており、一般の人でもそれらの衛星が送出する測位信号を利用して自身の位置を知ることができます。受信機では、多くの測位衛星を利用することで測位の精度を向上させています。ここでは、受信機内で行われる信号の処理や、より高精度な測位を行うための工夫について解説します。(編集部)

本稿で試す測位信号の抽出方法

● 各測位衛星に合わせて作るレプリカ信号を使う

それぞれの測位衛星からは、異なる拡散コード列が送信されています。GNSS受信機においては内部的に各測位衛星と同じ拡散コード列(レプリカ信号)を発生させ、相関処理を行うことによって、相関値ピークを捕捉および追尾します。

図1にGNSS受信機の追尾処理を簡易的にモデル化したブロック図を示します。コリレータを中心とした追尾ループとなっており、コード発生用数値制御発振器(コードNCO)によりタイミングを制御し、入力信号との相関処理を行うことで、スペクトル逆拡散(スペクトル拡散に用いた符号列を掛けて積算)を行って相関値を算出します。

なお、本稿では簡単のため最もシンプルなワイド・コリレータ(Early/Lateレプリカを ± 0.5 チップに配置)を解説します。高性能なGNSS受信機では、より

マルチパス耐性に優れたコリレータ(ナロー・コリレータなど)が搭載されています。

▶ コリレータは前後にずらしたレプリカ信号で相関ピークを追尾する

まず、相関ピークを追尾するためにタイミングが0.5チップ進んだEarlyレプリカ信号と、0.5チップ遅れたLateレプリカ信号を生成します。そして受信信号とレプリカ信号との相関値を算出します。それぞれのレプリカ信号との相関値の差がゼロになるタイミングに制御することで相関ピークを精度良く検出します(図2)。

▶ 航法データの復調

進みや遅延のないPromptレプリカ信号を生成して相関処理を行うとともに、位相同期回路(Phase-locked loop, PLL)においてキャリア弁別器の出力をキャリアNCOにフィードバックします。これにより入力信号との周波数差がゼロになる制御を行い、ドップラー信号の影響を除去した後、衛星位置の補正に用いる航法データを復調します。

シミュレーション1…受信機の内部処理

拡散コードによる測距の仕組みを知るために、衛星からのC/Aコード測位信号(民生用)をユーザ受信機で受信する簡単なシミュレーションをPythonで行ってみます。なお、ここではスペクトル拡散による測距

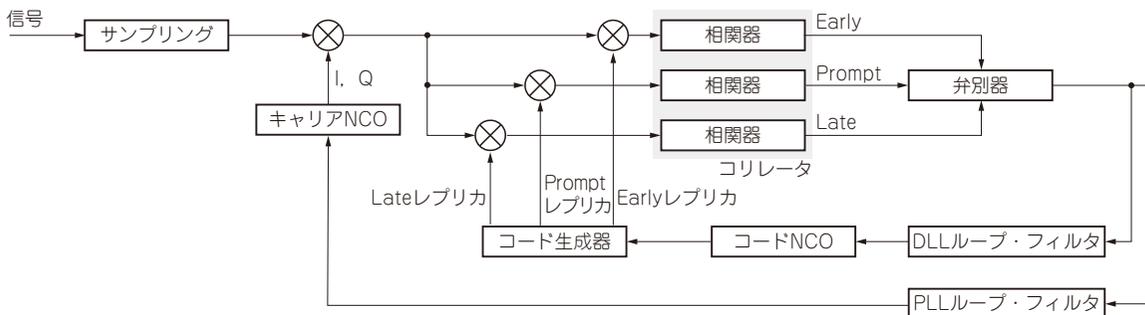


図1 GNSS受信機の追尾処理ブロック図