

Аналитический метод параметрической идентификации линейных динамических объектов

Симаньков В.И.

Научно-производственное общество с ограниченной ответственностью «ОКБ ТСП»

Описывается метод параметрической идентификации линейных моделей объектов автоматики по экспериментальным частотным характеристикам, отличающийся от описанных в классической литературе. Предложен двухэтапный метод параметрической идентификации по частотным характеристикам: сначала аппроксимация фазочастотной характеристики (ФЧХ) объекта, затем аппроксимация амплитудно-частотной характеристики (АЧХ) объекта.

Производится краткий обзор, сравнение и выбор численного метода решения задачи наименьших квадратов для аппроксимации ФЧХ объекта аналитическим выражением. На первом этапе идентификации рекомендуется применять метод сопряженных градиентов в модификации Полака-Рибьера, на втором этапе – алгебраический метод определения экстремума квадратичной функции. Производится параметрическая идентификация реального объекта управления по экспериментальным данным предложенным методом. Выполняется сравнение разработанного метода и метода “*Process models estimation*”, реализованного в САПР *MATLAB*.

Предложенный метод может использоваться для параметрической идентификации динамических объектов, и имеет преимущество в точности идентификации ФЧХ перед методом «*Process models estimation*» при незначительном снижении точности аппроксимации АЧХ и амплитудно-фазочастотной характеристики объекта. Достоинствами метода являются высокая скорость сходимости, сбалансированная точность аппроксимации частотных характеристик, сравнительная простота. Высокая скорость сходимости обусловлена тем, что точность вычислений градиента и целевой функции по аналитическим выражениям выше точности численной аппроксимации. Это позволяет делать точные шаги в точном направлении, что уменьшает количество итераций. Недостатком является сложность аналитических записей целевых функций и их градиентов, кусочно-заданные функции ФЧХ колебательных и форсирующих второго порядка звеньев. Это увеличивает вычислительную сложность алгоритма и снижает универсальность метода. Однако повышенная вычислительная сложность компенсируется снижением числа итераций и меньшим машинным временем.