Заметим, что в общем случае СИСТЕМА ДУ – это настолько убойное понятие, что какие-то алгоритмы для такого общего случая придумать невозможно. Поэтому и рассматривают только системы линейных ДУ, благо что они и попадаются гораздо чаще:

$$\begin{cases} \dot{x}(t) = a(t)x + b(t)y + X(t) \\ \dot{y}(t) = c(t)x + d(t)y + Y(t) \end{cases}$$

(все выкладки я буду проделывать для двух переменных, для бОльшего числа совершенно аналогично).

Наша цель – вывести алгоритм решения таких систем.

Сперва рассмотрим случай, когда коэфы явно не зависят от времени:

$$\begin{matrix} \dot{x}(t) \\ \dot{y}(t) \end{matrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{matrix} x(t) \\ y(t) \end{matrix} + \begin{matrix} X(t) \\ Y(t) \end{matrix}$$

И ещё пока предположим, что нет неоднородности:

$$\begin{matrix} \dot{x}(t) \\ \dot{y}(t) \end{matrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{matrix} x(t) \\ y(t) \end{matrix}$$

У нас же есть матрица $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$. У неё есть два собственных вектора: φ_1 (это столбец $\begin{matrix} p_1 \\ q_1 \end{matrix}$), и φ_2 (это столбец $\begin{matrix} p_2 \\ q_2 \end{matrix}$). Напомним, что это такие вектора, на которых матрица действует хорошо, т.е. проще всего – просто домножает на число:

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{matrix} p_1 \\ q_1 \end{matrix} = \lambda_1 \begin{matrix} p_1 \\ q_1 \end{matrix}$$

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{matrix} p_2 \\ q_2 \end{matrix} = \lambda_2 \begin{matrix} p_2 \\ q_2 \end{matrix}$$

Идея: искать решение в виде разложения на собственные векторы матрицы! Вот так:

$$\begin{matrix} \dot{x}(t) \\ \dot{y}(t) \end{matrix} = c_1(t) \begin{matrix} p_1 \\ q_1 \end{matrix} + c_2(t) \begin{matrix} p_2 \\ q_2 \end{matrix}$$

Подставляем в наше ДУ:

$$\begin{matrix} \dot{x}(t) \\ \dot{y}(t) \end{matrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{matrix} x(t) \\ y(t) \end{matrix}$$

И получаем $\dot{c_1}(t) \begin{matrix} p_1 \\ q_1 \end{matrix} + \dot{c_2}(t) \begin{matrix} p_2 \\ q_2 \end{matrix} = \lambda_1 c_1(t) \begin{matrix} p_1 \\ q_1 \end{matrix} + \lambda_2 c_2(t) \begin{matrix} p_2 \\ q_2 \end{matrix}$

откуда мгновенно

$$\begin{cases} \dot{c_1}(t) = \lambda_1 c_1(t) \\ \dot{c_2}(t) = \lambda_2 c_2(t) \end{cases}$$

Переменные разделились! Такое мы умеем решать:

$$\begin{cases} \dot{c_1}(t) = C_1 e^{\lambda_1 t} \\ \dot{c_2}(t) = C_2 e^{\lambda_2 t} \end{cases}$$

Константы С1 и С2 определяются из начальных условий.

Теперь давим внешнее возмущение:

$$\begin{matrix}\dot{x}(t) \\ \dot{y}(t)\end{matrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{matrix}x(t) \\ y(t)\end{matrix} + \begin{matrix}X(t) \\ Y(t)\end{matrix}$$

Тогда

$$\dot{c_1}(t)\begin{matrix}p_1 \\ q_1\end{matrix} + \dot{c_2}(t)\begin{matrix}p_2 \\ q_2\end{matrix} = \lambda_1 c_1(t)\begin{matrix}p_1 \\ q_1\end{matrix} + \lambda_2 c_2(t)\begin{matrix}p_2 \\ q_2\end{matrix} + \begin{matrix}X(t) \\ Y(t)\end{matrix}$$

Придётся и возмущение разложить на собственные вектора:

$$\begin{matrix}X(t) \\ Y(t)\end{matrix} = f_1(t)\begin{matrix}p_1 \\ q_1\end{matrix} + f_2(t)\begin{matrix}p_2 \\ q_2\end{matrix}$$

Вот теперь мы готовы:

$$\dot{c_1}(t)\begin{matrix}p_1 \\ q_1\end{matrix} + \dot{c_2}(t)\begin{matrix}p_2 \\ q_2\end{matrix} = \lambda_1 c_1(t)\begin{matrix}p_1 \\ q_1\end{matrix} + \lambda_2 c_2(t)\begin{matrix}p_2 \\ q_2\end{matrix} + f_1(t)\begin{matrix}p_1 \\ q_1\end{matrix} + f_2(t)\begin{matrix}p_2 \\ q_2\end{matrix}$$

И вновь система разваливается:

$$\begin{cases} \dot{c_1}(t) = \lambda_1 c_1(t) + f_1(t) \\ \dot{c_2}(t) = \lambda_2 c_2(t) + f_2(t) \end{cases}$$

А это мы решать также умеем – это ДУ на одну переменную!

орошо, а что делать в случае, когда матрица непостоянна во времени?

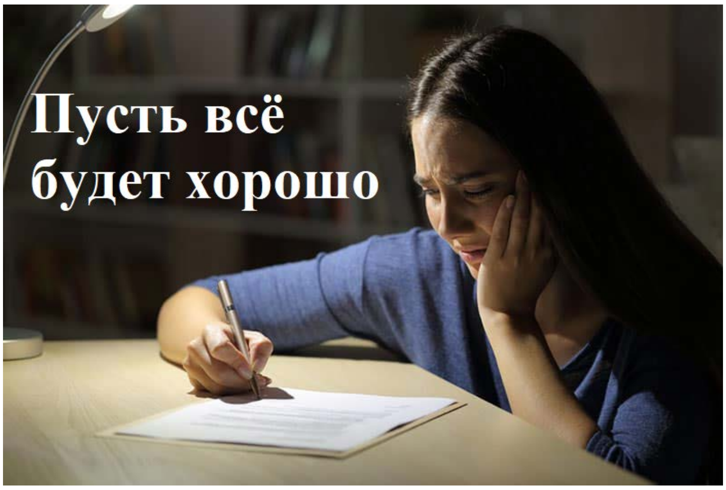
$$\begin{matrix}\dot{x}(t) \\ \dot{y}(t)\end{matrix} = \begin{pmatrix} a(t) & b(t) \\ c(t) & d(t) \end{pmatrix} \begin{matrix}x(t) \\ y(t)\end{matrix}$$

Собственные вектора и собственные значения у неё всё равно будут – вот только они теперь будут зависеть от времени.

$$\begin{pmatrix} a(t) & b(t) \\ c(t) & d(t) \end{pmatrix} \begin{pmatrix} p_1 \\ q_1 \end{pmatrix}(t) = \lambda_1(t) \begin{pmatrix} p_1 \\ q_1 \end{pmatrix}(t)$$

$$\begin{pmatrix} a(t) & b(t) \\ c(t) & d(t) \end{pmatrix} \begin{pmatrix} p_2 \\ q_2 \end{pmatrix}(t) = \lambda_2(t) \begin{pmatrix} p_2 \\ q_2 \end{pmatrix}(t)$$

Снова раскладываем решение на собственные вектора в надежде, что всё будет хорошо:



$$\begin{matrix} \dot{x}(t) \\ \dot{y}(t) \end{matrix} = c_1(t)\begin{matrix} p_1 \\ q_1 \end{matrix}(t) + c_2(t)\begin{matrix} p_2 \\ q_2 \end{matrix}(t)$$

К счастью, снова всё хорошо:

$$\dot{c_1}(t)\begin{matrix} p_1 \\ q_1 \end{matrix}(t) + \dot{c_2}(t)\begin{matrix} p_2 \\ q_2 \end{matrix}(t) = \lambda_1(t)c_1(t)\begin{matrix} p_1 \\ q_1 \end{matrix}(t) + \lambda_2 c_2(t)\begin{matrix} p_2 \\ q_2 \end{matrix}(t) + f_1(t)\begin{matrix} p_1 \\ q_1 \end{matrix}(t) + f_2(t)\begin{matrix} p_2 \\ q_2 \end{matrix}(t)$$

И всё-таки конец счастливый – вновь переменные разделяются

$$\begin{cases} \dot{c_1}(t) = \lambda_1(t)c_1(t) + f_1(t) \\ \dot{c_2}(t) = \lambda_2(t)c_2(t) + f_2(t) \end{cases}$$

что решается, например, вариацией постоянной.

Мне не очень хочется разбирать примеры – они и так у вас на семинарах.
Посыл изначально другой – я привёл простой (относительно) способ решать системы ДУ в том числе с переменными коэфами. А что кафмат?
А кафмат упарывается по полной, предлагая какое-то безумное решение через матрицу Коши (матрицант). Это всё стремление запихнуть всё в одну формулу. Да, она относительно короткая, но ею невозможно пользоваться.
Автору приходилось решать много линейных систем ДУ и всякий раз он решал через алгоритм разложения по собственным векторам. Он простой и легко его запомнить. А убер-формулы с умножением матриц от кафмата… фу ☺