Linear differential-algebraic systems with selected unknowns

Anton A. Panferov

CMC MSU

March 30, 2016

Differential systems with selected unknowns

Let K be a differential field of characteristic 0 with the derivation $\partial = '$.

We consider a system of ordinary differential equations

$$L(y) = 0, \tag{1}$$

where $L \in K[\partial]^{m \times m}$ is a differential operator of full rank,
$y = (y_1, \dots, y_m)^T$ is a vector of unknowns.

We assume that a part of unknowns (components of the vector y) is of more interest to us than the other unknowns. We call these components **selected** ones.

We can address a number of problems:

- the check for existence of the solutions whose selected components belong to given classes;
- the search for selected solution components only;
- the check for partial stability over selected components etc.

Contents

AB-algorithm

Consider a *normal* differential system of the form

$$y' = Ay, \tag{2}$$

where $y = (y_1, \ldots, y_m)^T$ is a vector of unknowns, $A \in K^{m \times m}$.

For systems of the form (2) S. A. Abramov and M. Bronstein proposed an algorithm (AB-algorithm) that, for the selected components of the unknown vector, makes it possible to turn to the normal system

$$z' = Bz, \tag{3}$$

where the components of z are the selected components of y and, possibly, some their derivatives.

AB-algorithm

$$y' = Ay \quad \Longrightarrow \quad z' = Bz$$

1. The projections of the solution space on selected unknowns in arbitrary differential extension of the initial differential field of the initial system $y' = Ay$ and the system $z' = Bz$ are identical.
2. If the solution to the system $z' = Bz$ is such that its selected components belong to some differential extension of the initial differential field, then all the components of this solution belong to this extension.
3. If the size of B is equal to the size of A and the initial system has a solution whose selected components belong to some differential extension of the initial differential field, then *all* the components of this solution belong to this extension.

AB-algorithm: example

AB-algorithm is implemented in MAPLE as procedure `ReducedSystem` that is a part of the standard package `OreTools`.

Example: $y = (y_1, \boldsymbol{y_2}, y_3)^T$ $\boldsymbol{y_2}$ is selected

$$y' = \begin{bmatrix} 1 & -2 & -1 \\ 1 & -(2x+1)/x & -1 \\ -1 & 2(x+1)/x & 1 \end{bmatrix} y \Longrightarrow z' = \begin{bmatrix} 0 & 1 \\ 1/x^2 & -1/x \end{bmatrix} z, \quad z = (\boldsymbol{y_2}, \boldsymbol{y_2'})^T$$

Higher order systems

$$A_r y^{(r)} + A_{r-1} y^{(r-1)} + \ldots + A_1 y' + A_0 y = 0, \qquad (4)$$
$$A_i \in K^{m \times m}, y = (y_1, \ldots, y_m)^T$$

Higher order systems

$$A_r y^{(r)} + A_{r-1} y^{(r-1)} + \ldots + A_1 y' + A_0 y = 0, \tag{4}$$

$$A_i \in K^{m\times m}, y = (y_1, \ldots, y_m)^T$$

$$\Downarrow A_r \text{ is invertible}$$

$$Y' = \begin{bmatrix} 0 & I_m & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & I_m \\ -A_r^{-1}A_0 & -A_r^{-1}A_1 & \ldots & -A_r^{-1}A_{r-1} \end{bmatrix} Y$$

where I_m is identity $m \times m$ matrix,

$$Y = \left(y_1, \ldots, y_m, y_1', \ldots, y_m', \ldots, y_1^{(r-1)}, \ldots, y_m^{(r-1)}\right)^T$$

Higher order systems

$$A_r y^{(r)} + A_{r-1} y^{(r-1)} + \ldots + A_1 y' + A_0 y = 0, \quad (4)$$
$$A_i \in K^{m\times m}, y = (y_1, \ldots, y_m)^T$$
$$\Downarrow$$
$$\begin{bmatrix} I_m & & & 0 \\ & I_m & & \\ & & \ddots & \\ 0 & & & A_r \end{bmatrix} Y' + \begin{bmatrix} 0 & -I_m & & 0 \\ & \ddots & \ddots & \\ 0 & & 0 & -I_m \\ A_0 & A_1 & \ldots & A_{r-1} \end{bmatrix} Y = 0$$

where I_m is identity $m \times m$ matrix,

$$Y = \left(y_1, \ldots, y_m, y_1', \ldots, y_m', \ldots, y_1^{(r-1)}, \ldots, y_m^{(r-1)}\right)^T$$

Extract algorithm

Consider a differential system of the form

$$A_1 y' + A_0 y = 0, \tag{5}$$

where $A_1, A_0 \in K^{m\times m}$ are leading and trailing matrices,
$y = (y_1, \ldots, y_m)^T$ is unknown vector, some components of which are *selected*.

Suppose $A_1 \not\equiv 0$ and $\det A_1 \equiv 0$.

Such systems are called **differential-algebraic** systems.

For such systems the `Extract` algorithm produces the normal differential system

$$\tilde{y}' = A\tilde{y} \tag{6}$$

for the part of components of y ($\tilde{y} \subset y$).
The selected components of y that are not the part of $\tilde{y}$ are linearly expressed only via the selected unknowns from $\tilde{y}$.

Extract algorithm

Input: the differential-algebraic system $A_1 y' + A_0 y = 0$ in a row-reduced form and the set of the selected unknowns.

The algorithm consists of three stages:

1. elimination of unselected unknowns (from the differential part of the system);
2. elimination of selected unknowns;
3. expression of the eliminated selected unknowns via the selected unknowns remained in the differential system.

Output: matrices of the new differential system and the algebraic system.

Extract algorithm

$$S\colon\ A_1 y' + A_0 y = 0$$

$S_d\colon\ \tilde{y}' = A\tilde{y}$

$(\tilde{y} \subset y)$

S_a: algebraic system

(expressions of the selected components of y, that are not the part of $\tilde{y}$)

Extract algorithm

$$S\colon\ A_1 y' + A_0 y = 0$$

selected unknowns: s

$$S_d\colon\ \tilde{y}' = A\tilde{y} \quad (\tilde{y} \subset y)$$

S_a: algebraic system

Extract algorithm

$$S\colon\ A_1 y' + A_0 y = 0$$

selected unknowns: $s = s_1 \cup s_2$ $(s_1 \cap s_2 = \varnothing)$

(s_1) → $S_d\colon\ \tilde{y}' = A\tilde{y}$ $(\tilde{y} \subset y)$

(s_2) → S_a: algebraic system

$$y_{s_2} = R\, y_{s_1} \tag{7}$$

R is a matrix over K

Extract algorithm

$$S\colon\ A_1 y' + A_0 y = 0$$

selected unknowns: $s = s_1 \cup s_2$ $(s_1 \cap s_2 = \varnothing)$

(s_1)

$$S_d\colon\ \tilde{y}' = A\tilde{y} \quad (\tilde{y} \subset y)$$

(s_2)

S_a: algebraic system

$$y_{s_2} = R\, y_{s_1} \tag{7}$$

R is a matrix over K

Definition

The systems S_d, S_a are said to be *consistent* with (S, s), if the projection of the solution space of S on s coincides with the projection of the solution space of S_d, S_a on s in arbitrary differential extension of the initial differential field.

Extract algorithm

$$S\colon\ A_1 y' + A_0 y = 0$$

selected unknowns: $s = s_1 \cup s_2$ $(s_1 \cap s_2 = \varnothing)$

(s_1)

$$S_d\colon\ \tilde{y}' = A\tilde{y} \quad (\tilde{y} \subset y)$$

(s_2)

S_a: algebraic system

$$y_{s_2} = R\, y_{s_1} \tag{7}$$

R is a matrix over K

Definition

The systems S_d, S_a are said to be *consistent* with (S, s), if the projection of the solution space of S on s coincides with the projection of the solution space of S_d, S_a on s in arbitrary differential extension of the initial differential field.

Proposition

Let S_d, S_a be consistent with (S, s). Then the size of S_a is uniquely determined only by the initial system S and the set of the selected unknowns s.

Example

$K = \mathbb{Q}(x), \quad \partial = \frac{d}{dx}$

$$\begin{bmatrix} x & 1 & 0 \\ x^2 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} y' + \begin{bmatrix} -1 & -1 & 0 \\ -1 & 0 & 0 \\ -x & 0 & 1 \end{bmatrix} y = 0,$$

where $y = (\boldsymbol{y_1}, \boldsymbol{y_2}, \boldsymbol{y_3})^T$. All unknowns are selected.

1. The result of Extract:

$$S_d: \tilde{y}' = \begin{bmatrix} 1/x & 0 \\ 0 & 1 \end{bmatrix} \tilde{y}, \quad \tilde{y} = (\boldsymbol{y_1}, \boldsymbol{y_2})^T$$
$$S_a: \boldsymbol{y_3} = x\,\boldsymbol{y_1}$$

2. Inconsistent systems to get rational solutions:

$$S_d: \boldsymbol{y_1'} = \frac{1}{x}\boldsymbol{y_1} \qquad S_a: \begin{cases} \boldsymbol{y_2} = 0 \\ \boldsymbol{y_3} = x\,\boldsymbol{y_1} \end{cases}$$

Example

$K = \mathbb{Q}(x), \quad \partial = \frac{d}{dx}$

$$\begin{bmatrix} x & 1 & 0 \\ x^2 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} y' + \begin{bmatrix} -1 & -1 & 0 \\ -1 & 0 & 0 \\ -x & 0 & 1 \end{bmatrix} y = 0,$$

where $y = (\boldsymbol{y_1}, \boldsymbol{y_2}, \boldsymbol{y_3})^T$. All unknowns are selected.

1. The result of Extract:

$$S_d : \tilde{y}' = \begin{bmatrix} 1/x & 0 \\ 0 & 1 \end{bmatrix} \tilde{y}, \quad \tilde{y} = (\boldsymbol{y_1}, \boldsymbol{y_2})^T$$
$$S_a : \boldsymbol{y_3} = x\, \boldsymbol{y_1}$$

2. Inconsistent systems to get rational solutions:

$$S_d : \boldsymbol{y_1'} = \frac{1}{x} \boldsymbol{y_1} \qquad S_a : \begin{cases} \boldsymbol{y_2} = 0 \\ \boldsymbol{y_3} = x\, \boldsymbol{y_1} \end{cases}$$

Extract algorithm

$$S\colon\ A_1 y' + A_0 y = 0$$

$$S_d\colon\ \tilde{y}' = A\tilde{y} \quad (\tilde{y} \subset y)$$

S_a: algebraic system

(expressions of the selected components of y, that are not the part of $\tilde{y}$)

For the given differential-algebraic system S and the set of the selected unknowns s there are infinite number of pairs of consistent systems S_d, S_a. The size of S_d is unbounded.

Extract algorithm

$$S\colon\ A_1 y' + A_0 y = 0$$

S_d: $\tilde{y}' = A\tilde{y}$
$(\tilde{y} \subset y)$

S_a: algebraic system
(expressions of the selected components of y, that are not the part of $\tilde{y}$)

For the given differential-algebraic system S and the set of the selected unknowns s there are infinite number of pairs of consistent systems S_d, S_a. The size of S_d is unbounded.

Extract: example

Suppose S_d is a differential system constructed by the Extract algorithm; then the size of S_d is not always minimal.

$$S: \begin{bmatrix} x & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} y' + \begin{bmatrix} 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -x & 0 & 0 & 1 \end{bmatrix} y = 0$$

$$y = (\boldsymbol{y_1}, \boldsymbol{y_2}, y_3, y_4)^T, \quad y_1, y_2 \text{ are selected}$$

$$S_d: \tilde{y}' = \begin{bmatrix} -1/x & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} \tilde{y}, \quad \tilde{y} = (\boldsymbol{y_1}, \boldsymbol{y_2}, y_3)^T$$

$$S_a: \varnothing$$

Extract: example

Suppose S_d is a differential system constructed by the Extract algorithm; then the size of S_d is not always minimal.

$$S: \begin{bmatrix} x & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} y' + \begin{bmatrix} 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -x & 0 & 0 & 1 \end{bmatrix} y = 0$$

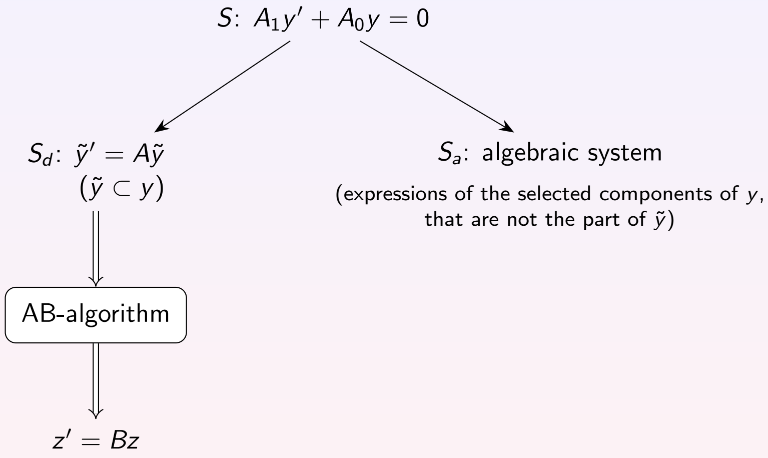
$$y = (\boldsymbol{y_1}, \boldsymbol{y_2}, y_3, y_4)^T, \quad y_1, y_2 \text{ are selected}$$

$$S_d: \tilde{y}' = \begin{bmatrix} -1/x & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} \tilde{y}, \quad \tilde{y} = (\boldsymbol{y_1}, \boldsymbol{y_2}, y_3)^T$$

$$S_a: \varnothing$$

$$\tilde{y}' = \begin{bmatrix} -1/x & 0 \\ 1 & 0 \end{bmatrix} \tilde{y}, \quad \tilde{y} = (\boldsymbol{y_1}, \boldsymbol{y_2})^T$$

ExtrAB = Extract + AB-algorithm



ExtrAB algorithm

$$S\colon\ A_1 y' + A_0 y = 0$$

$$S_d^{\mathrm{AB}}\colon\ z' = Bz$$

(z is a part of the selected unknowns of y and some their derivatives)

$$S_a\colon\ \text{algebraic system}$$

(expressions of the selected components of y, that are not the part of z)

ExtrAB algorithm

$$S\colon\ A_1 y' + A_0 y = 0$$

$S_d^{\mathrm{AB}}\colon\ z' = Bz$

(z is a part of the selected unknowns of y and some their derivatives)

S_a: algebraic system

(expressions of the selected components of y, that are not the part of z)

Theorem

The systems S_d^{AB} and S_a produced by ExtrAB algorithm

1. are consistent with (S, s);
2. have the minimal sizes.

ExtrAB algorithm: example

$$\begin{bmatrix} -1 & 0 & -x & 0 & x & 0 \\ 0 & 0 & (x+1)x & 0 & 0 & x \\ 0 & 0 & x & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} y' + \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & -1 & 0 & -(x+1) \\ 0 & 0 & 0 & -1 & 0 & -1 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ -1 & 0 & 0 & 1 & 0 & 1 \\ 1 & x & 0 & 0 & 0 & 0 \end{bmatrix} y = 0$$

$$y = (\boldsymbol{y_1}, \boldsymbol{y_2}, y_3, y_4, y_5, y_6)^T, \quad y_1, y_2 \text{ are selected}$$

This system does not have Laurent series solutions with nonzero y_1, y_2. At the same time it has solutions where y_1, y_2 are nonzero Laurent series. We will show how to use ExtrAB algorithm to find them.

ExtrAB algorithm: example

$$\begin{bmatrix} -1 & 0 & -x & 0 & x & 0 \\ 0 & 0 & (x+1)x & 0 & 0 & x \\ 0 & 0 & x & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} y' + \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & -1 & 0 & -(x+1) \\ 0 & 0 & 0 & -1 & 0 & -1 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ -1 & 0 & 0 & 1 & 0 & 1 \\ 1 & x & 0 & 0 & 0 & 0 \end{bmatrix} y = 0$$

$$y = (\boldsymbol{y_1}, \boldsymbol{y_2}, y_3, y_4, y_5, y_6)^T, \quad y_1, y_2 \text{ are selected}$$

Step 1: Extract

$$S_d\colon \tilde{y}' = \begin{bmatrix} 1-1/x & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -1/x & 0 \\ 1 & 0 & x+1 & 1 \end{bmatrix} \tilde{y}, \quad \tilde{y} = (\boldsymbol{y_2}, y_3, y_5, y_6)^T$$

$$S_a\colon \boldsymbol{y_1} = -x\, \boldsymbol{y_2}$$

ExtrAB algorithm: example

$$\begin{bmatrix} -1 & 0 & -x & 0 & x & 0 \\ 0 & 0 & (x+1)x & 0 & 0 & x \\ 0 & 0 & x & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} y' + \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & -1 & 0 & -(x+1) \\ 0 & 0 & 0 & -1 & 0 & -1 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ -1 & 0 & 0 & 1 & 0 & 1 \\ 1 & x & 0 & 0 & 0 & 0 \end{bmatrix} y = 0$$

$$y = (\boldsymbol{y_1}, \boldsymbol{y_2}, y_3, y_4, y_5, y_6)^T, \quad y_1, y_2 \text{ are selected}$$

Step 1: Extract

$$S_d\colon \tilde{y}' = \begin{bmatrix} 1 - 1/x & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -1/x & 0 \\ 1 & 0 & x+1 & 1 \end{bmatrix} \tilde{y}, \quad \tilde{y} = (\boldsymbol{y_2}, y_3, y_5, y_6)^T$$

$$S_a\colon \boldsymbol{y_1} = -x\, \boldsymbol{y_2}$$

Step 2: AB-algorithm

$$S_d \Longrightarrow S_d^{\mathrm{AB}}\colon z' = \begin{bmatrix} 0 & 1 \\ 1/x & 1 - 2/x \end{bmatrix} z, \quad z = (\boldsymbol{y_2}, \boldsymbol{y_2'})^T$$

ExtrAB algorithm: example

$$\begin{bmatrix} -1 & 0 & -x & 0 & x & 0 \\ 0 & 0 & (x+1)x & 0 & 0 & x \\ 0 & 0 & x & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} y' + \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & -1 & 0 & -(x+1) \\ 0 & 0 & 0 & -1 & 0 & -1 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ -1 & 0 & 0 & 1 & 0 & 1 \\ 1 & x & 0 & 0 & 0 & 0 \end{bmatrix} y = 0$$

$$y = (\boldsymbol{y_1}, \boldsymbol{y_2}, y_3, y_4, y_5, y_6)^T, \quad y_1, y_2 \text{ are selected}$$

Produced systems:

$$S_d^{\mathrm{AB}}\colon z' = \begin{bmatrix} 0 & 1 \\ 1/x & 1 - 2/x \end{bmatrix} z, \quad z = (\boldsymbol{y_2}, \boldsymbol{y_2'})^T \qquad S_a\colon \boldsymbol{y_1} = -x\, \boldsymbol{y_2}$$

ExtrAB algorithm: example

$$\begin{bmatrix} -1 & 0 & -x & 0 & x & 0 \\ 0 & 0 & (x+1)x & 0 & 0 & x \\ 0 & 0 & x & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} y' + \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & -1 & 0 & -(x+1) \\ 0 & 0 & 0 & -1 & 0 & -1 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ -1 & 0 & 0 & 1 & 0 & 1 \\ 1 & x & 0 & 0 & 0 & 0 \end{bmatrix} y = 0$$

$$y = (\boldsymbol{y_1}, \boldsymbol{y_2}, y_3, y_4, y_5, y_6)^T, \quad y_1, y_2 \text{ are selected}$$

Produced systems:

$$S_d^{\mathrm{AB}}:\ z' = \begin{bmatrix} 0 & 1 \\ 1/x & 1 - 2/x \end{bmatrix} z, \quad z = (\boldsymbol{y_2}, \boldsymbol{y_2'})^T \qquad S_a:\ \boldsymbol{y_1} = -x\,\boldsymbol{y_2}$$

Rational solutions for y_1, y_2:

$$S_d^{\mathrm{AB}} \Rightarrow \boldsymbol{y_2} = C/x, \qquad S_a \Rightarrow \boldsymbol{y_1} = C$$

ExtrAB algorithm: example

$$\begin{bmatrix} -1 & 0 & -x & 0 & x & 0 \\ 0 & 0 & (x+1)x & 0 & 0 & x \\ 0 & 0 & x & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} y' + \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & -1 & 0 & -(x+1) \\ 0 & 0 & 0 & -1 & 0 & -1 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ -1 & 0 & 0 & 1 & 0 & 1 \\ 1 & x & 0 & 0 & 0 & 0 \end{bmatrix} y = 0$$

$$y = (\boldsymbol{y_1}, \boldsymbol{y_2}, y_3, y_4, y_5, y_6)^T, \quad y_1, y_2 \text{ are selected}$$

Produced systems:

$$S_d^{\mathrm{AB}}: z' = \begin{bmatrix} 0 & 1 \\ 1/x & 1 - 2/x \end{bmatrix} z, \quad z = (\boldsymbol{y_2}, \boldsymbol{y_2'})^T \qquad S_a: \boldsymbol{y_1} = -x\, \boldsymbol{y_2}$$

Rational solutions for y_1, y_2:

$$S_d^{\mathrm{AB}} \Rightarrow \boldsymbol{y_2} = C/x, \qquad S_a \Rightarrow \boldsymbol{y_1} = C$$

Laurent series solutions for y_1, y_2:

$$S_d^{\mathrm{AB}} \Rightarrow \boldsymbol{y_2} = (C_1 e^x + C_2)/x, \qquad S_a \Rightarrow \boldsymbol{y_1} = C_1 e^x + C_2$$

Implementation (http://www.ccas.ru/ca/extract)

$$\begin{bmatrix} -1 & 0 & -x & 0 & x & 0 \\ 0 & 0 & (x+1)x & 0 & 0 & x \\ 0 & 0 & x & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} y' + \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & -1 & 0 & -(x+1) \\ 0 & 0 & 0 & -1 & 0 & -1 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ -1 & 0 & 0 & 1 & 0 & 1 \\ 1 & x & 0 & 0 & 0 & 0 \end{bmatrix} y = 0$$

$$y = (\boldsymbol{y_1}, \boldsymbol{y_2}, y_3, y_4, y_5, y_6)^T, \quad y_1, y_2 \text{ are selected}$$

```
> Extract(A1, A0, {1,2}, R)
```

$$\begin{bmatrix} 1-1/x & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -1/x & 0 \\ 1 & 0 & x+1 & 1 \end{bmatrix}, \{[2,1]\}, [-x], \{[1,1]\}$$

```
> ReducedSystem(%[1], {1}, R)
```

$$\left[\begin{bmatrix} 0 & 1 \\ 1/x & 1-2/x \end{bmatrix}, \{[1,1]\} \right]$$