IEooc_Methods6_Exercise2: Tracing resources through input-output tables

Goal: *Understand the differences between Leontief input-output (IO), Leontief price, Ghosh IO and absorbing Markov Chain models. Learn how they all can be used to determine the distribution of natural resource or value added input into different final demand sectors (so-called end-use shares). Apply the resulting equations to a test IO table.*

Core reading for this exercise:

IEooc_Methods6_Reading4: Resource tracing with input output (IO) models – an overview.

End-use shares with IO models. Input output (IO) models are used to answer different research questions. Next to the questions on product environmental footprints (answered with the environmentally extended Leontief IO model) and commodity prices or material composition (answered with Leontief price models), the allocation of resource input to final demand sectors, the so-called sector split or, as synonym, end-use shares, is an important application.

This type of question is complementary to the footprint question. For footprints, the supply chain of a final demand category is constructed so that different final commodities are linked to multiple upstream resources. For the resource allocation, different resource inputs are traced into multiple downstream commodities.

In the core reading for this work (IEooc_Methods6_Reading4: Resource tracing with input output (IO) models – an overview), we show that the allocation matrix of resources to final products, the so-called end-use shares, sector split or *D*-matrix, is the same for three different IO models, the Leontief quantity, the Leontief price, and the Ghosh model.

This finding holds when considering value added as exogenous input to the economy and when working at scale, i.e., performing the calculations for the full inventoried value added and final demand. In the reading and the workbook for this exercise, we also show that the D_{res}-matrix is the same for all three approaches, when considering natural (material) resources as exogenous input to the economy.

In the workbook for this exercise, IEooc_Methods6_Exercise2_Resource_tracing_IO_Workbook.xlsx, the sector split or end-use share matrix *D* is calculated for value added allocation (sheet *Allocate_v_to_y, determine D*) and D_{res} for natural resource input allocation (sheet *Allocate_b_to_y, determ. D_res*).

Questions and tasks:

On sheet *Allocate_b_to_y, determ. D_res*:

1) Check the matrix D_{res} calculated from the IO table: What is the meaning of its rows and colums? What summation constraint (row or column sum) applies to D_{res} and does it hold for the calculated example?

2) A major problem in using IO tables to trace resources is that the end-use sectors of the table must align with the end-use sectors of the material flow analysis (MFA) study for which the end-use shares are calculated. A typical situation is that packaging is an end-use category in the MFA study but not in the IO table, where all packaging material flows follow the product they contain.

   Here, the task is to simulate a packaging final demand category, by re-routing flows in the IO table so that flows of packaging material do not flow into the manufacturing sector where they are needed to package goods but flow into their own final demand sector instead.

   Assume, for this task, that sector no. 3 is the packaging sector.

   Re-route the use of packaging by other sectors into final demand.

   Re-balance the IO table, calculate D, and compare!

3) Another situation occurs when a product flow that goes into final demand should be located in another product group in the MFA study. Here, we consider a machinery flow that goes into final demand but that should be part of buildings instead to be consistent with the MFA study.

   Assume, for this task, that sector no. 1 is the machinery sector and sector no. 4 is the building sector.

   Re-route 1/3 of final machinery demand to buildings!

   Re-balance IO table, calculate D, and compare!

Hints:

Best create a copy of sheet *Allocate_b_to_y, determ. D_res* for tasks 2 and 3.