# Transportation Shareweights

## Files

* Reference shareweights: subsector\_shareweights\_ref.csv
  + From Model Interface reference run
* Default XML: transportation\_Subregions\_Thailand.xml
* Add-on XML 1: freight\_rail\_0034\_2040\_lin.xml
  + Linear interpolation from default 2015 value to 0.004 in 2040 for freight rail
* Add-on XML 2: freight\_road\_55\_2040\_lin.xml
  + Linear interpolation from default 2015 value (1) to 0.5 for freight road
* Batch file: batch.xml

## Goal

We are trying to match Thailand’s 2040 goals by decreasing freight service output by road and increasing freight service output by domestic ship and rail. In 2040, Thailand’s goal is for ship to account for 19% of freight service output and for rail to account for 10%.

## Approach

To do this, we will gradually decrease the shareweight for road through 2040 and also gradually increase the rail shareweight (only decreasing the road shareweight results in a ratio of ship to rail that is too high). From the reference shareweights file, we see that within the trn\_freight sector, the road subsector has a shareweight of 1 throughout all periods and the Freight Rail subsector has a shareweight of 0.001342 throughout all periods. Through experimentation, we’ve found that linearly increasing/decreasing shareweights to the following values in 2040 produces the desired outcome:

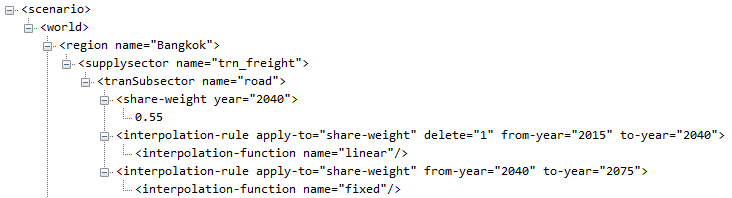
* road: 0.55
* Freight Rail: 0.0034

## GCAM Implementation

To implement these shareweight modifications in GCAM, we produced two add-on XMLs corresponding to the two subsectors we are modifying (road and Freight Rail). Within the appropriate tranSubsector tag, there are three tags we need to include:

* The shareweight in 2040
* The interpolation rule from 2015 to 2040 (linear)
* The interpolation rule from 2040 to 2075 (fixed)

Here is the structure of these tags:



\*\*\*Note that the first interpolation rule includes delete=”1”. This is necessary in order to override all of the previous (default) interpolation rules.

Since we are implementing “national” policies, we will just assume that the shareweight trajectories are the same for all subregions. Thus, the add-on XMLs have the same values copied into each subregion (Bangkok, Nonthaburi, Samut Prakan, and Rest of Thailand). See freight\_rail\_0034\_2040\_lin.xml and freight\_road\_55\_2040\_lin.xml for the final add-on files.

## Extra Information

### Batch Runs

To determine the best 2040 shareweights that produce the desired shares of freight service, we ran several batches of different shareweight combinations. The batch.xml file gives an example of the setup for running one such batch. Here is the [guide for setting up batch runs on PIC](https://docs.google.com/document/d/1aURCiSaKkbDFUSLpCnT7K5uNyws5ThxIQMr7E3bIWWc/edit#heading=h.cssp0x15n55k).

# EV Cost Parity

## Files

* Reference input costs: L254.StubTranTechCost\_trn\_Subregions\_Thailand
  + This is a gcamdata intermediate output and has the same values as the XML below
* Default XML: transportation\_Subregions\_Thailand.xml
* Cost calculation: Thailand\_ev\_cost.xlsx
* Add-on XML: trn\_cost\_ev\_2060p\_2070f.xml

## Goal

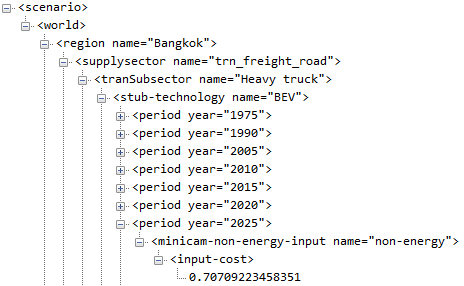
We are trying to increase EV market penetration to reflect Thailand’s plans to promote EV manufacturing and use.

## Approach

To do this, we will implement cost parity with non-EVs; i.e. simulate the input costs of EVs decreasing gradually until they match the input costs of non-EVs in a certain “cost parity year.” The example here assumes that passenger EVs will reach cost parity with passenger non-EVs in 2060 and freight EVs will reach cost parity with freight non-EVs in 2070. To calculate the EV cost trajectories, we set the 2060 (2070) input cost for the BEV and Hybrid Liquids technologies equal to the default 2060 (2070) input cost for the Liquids technology and use the compound annual growth rate to determine costs up to the parity year. See the thailand\_ev\_cost.xlsx file for details. Note that we need to do this calculation separately for each tranSubsector (e.g. Car, Large Car and Truck, etc.)

## GCAM Implementation

To implement these input cost modifications in GCAM, we produced an add-on XML. Within each relevant stub-technology (BEV and Hybrid Liquids) for each relevant tranSubSector (Heavy Truck, Light Truck, Medium truck, 2W and 3W, Car, Large Car and Truck, Mini Car), we set the desired input cost for each year. Note that we only modified the costs between 2020 and the parity year.



## Extra Information

### Generating XMLs

The R script generate\_trn\_cost\_xml.R has a function to calculate the input cost trajectories (rather than doing this in Excel) and also to generate the add-on XML using XML functions from the gcamdata package.