

Modell över Sveriges primärenergitillförsel
Kurs ENM155

Andreas Hagesjö Daniel Pettersson Magnus Hagmar
Niclas Ogeryd Robert Nyquist

24 november 2014

1 Introduktion

Denna rapport innehåller en enkel modell utav Sveriges energisystem som det ser ut idag. Modellen visar hur olika primärenergier fördelas på de tre sektorerna industri, transport och bostäder samt en uppskattning utav Sveriges totala primärenergitillförsel.

2 Metod

Modellen är nerbruten i tre delar, industri, transport och bostäder som är de olika sektorerna. För varje sektor så listas alla primärenergier som bidrar till respektive sektor. Varje primärenergi går sedan vidare till de olika sekundärenergierna som den bidrar till. Varje sekundärenergi går vidare till sektorn, alternativ till en ny sekundärenergi som i sin tur går vidare till en sektor eller ytterligare en sekundärenergi.

- Då vi har brutit ner modellen i sektorer så följer de inte diagrammet i Figur 1 i lab PM. Istället så ger flödesschemat i Appendix A en direkt bild utav strukturen på våran implementation.
- Modellen är byggd så att det går att ta reda på tillförseln av varje enskild primärenergi samt vilka typer av primärenergi, och mängden, varje enskild sektor använder. Det går även att räkna ut värden på sekundärenergierna för varje sektor med hjälp av modellen.
- Då varje sektor innehåller alla primärenergier och sekundärenergier som bidrar så blir det väldigt enkelt att addera nya energier. Den nya energin läggs till i sektorn den bidrar till och går sedan vidare till en sekundärenergi eller sektorn.

2.1 Matematisk modell

Vi definierar ett uttryck för att beräkna elen i transport samt industrisektorn, som vi sedan referenserar för att minska längden på det totala uttrycket.

I våra ekvationer betyder E_{fe} energin från fossila bränslen omvandlat till el. Samma sätt med E_{fjv} , fjärrvärme till värme, E_{ffj} , fossil till fjärrvärme etc.

2.2 Bostäder

$$EL = \frac{\frac{E_v k_{ev}}{\varphi_{ev}} + \frac{E_v k_{fjv} k_{vp}}{\varphi_{fjv} \varphi_{vp} \varphi_{trans}} + E_e}{\varphi_e \varphi_{trans}}$$

Fossila bränslen

$$E_f = \frac{E_v k_{fv}}{\varphi_{fv}} + \frac{E_v k_{fjv} k_{ffj}}{\varphi_{fjv} \varphi_{ffj} \varphi_{trans}} + EL \cdot \frac{k_{fe}}{\varphi_{fe}}$$

Biobränslen

$$E_b = \frac{E_v k_{bv}}{\varphi_{bv}} + \frac{E_v k_{fjv} k_{bfj}}{\varphi_{fjv} \varphi_{bfj} \varphi_{trans}} + EL \cdot \frac{k_{be}}{\varphi_{be}}$$

Vindkraft

$$E_{vind} = EL \cdot \frac{k_{vind}}{\varphi_{vind}}$$

Vattenkraft

$$E_{vatten} = EL \cdot \frac{k_{vatten}}{\varphi_{vatten}}$$

Kärnkraft

$$E_{karn} = EL \cdot \frac{k_{karn}}{\varphi_{karn}}$$

Spillvärme (inte en riktig energi, men måste tas med i beräkningarna)

$$E_{spill} = EL \cdot \frac{k_{spill}}{\varphi_{spill}}$$

2.3 Transport

$$E_f = \frac{E_t k_f}{\varphi_f \varphi_{drivmedel}}$$

$$E_b = \frac{E_t k_b}{\varphi_b \varphi_{drivmedel}}$$

2.4 Industri

Industrisektorn skiljer sig från bostadssektorn genom att det inte finns någon eluppvärmning, annars är de två sektorerna lika.

$$EL = \frac{\frac{E_v k_{ev}}{\varphi_{ev}} + E_e}{\varphi_e \varphi_{trans}}$$

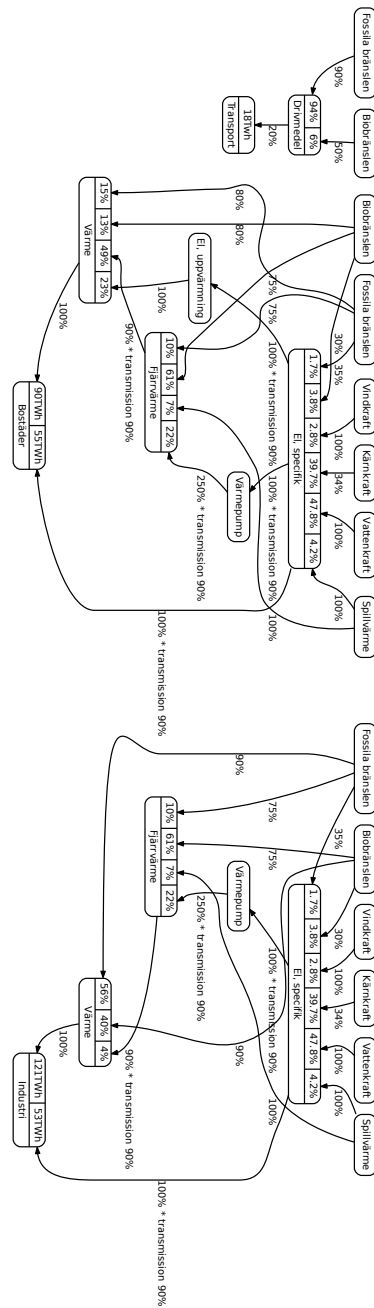
3 Resultat

I tabell 1 visas den totala energitillförseln samt varje enskild energikällas tillförsel.

Energikälla	Tillförsel
Fossila bränslen	201.483 TWh
Biobränsle	147.288 TWh
Vindkraft	4.187 TWh
Vattenkraft	71.491 TWh
Kärnkraft	174.638 TWh
Totalt	649.561 TWh

Tabell 1: Resultat

A Flödesschema



Figur 1: Flödesschema över energianvändning i Sverige.

B Programkod

I denna fil finns klasserna för sektorer samt energier.

```
class Sector:
    def __init__(self, name):
        self.name = name
        self.energies = {}

    def add_energy(self, name, energy):
        self.energies[name] = energy

    def value(self):
        return sum([self.energies[e].value() for e in
                    self.energies])

class Energy:
    def __init__(self, name, energy=0, quota=0,
                efficiency=0):
        self.name = name
        self.sectors = {}
        self.inputs = {}
        self.energy = energy
        self.quota = quota
        self.efficiency = efficiency

    def add_input(self, name, energy=0, quota=0,
                 efficiency=0):
        self.inputs[name] = Energy(name, energy, quota
                                    , efficiency)

    def add_sector(self, name):
        self.sectors[name] = Sector(name)
        self.sectors[name].add_energy(self.name, self)

    def value(self, sector=None):
        if sector:
            return self.sectors[sector].value()
        else:
            return (sum([self.inputs[i].value() for i
                        in self.inputs]) + self.energy) * self.
                    quota / self.efficiency
```

Här är själva programmet.

```
from Energy import Sector, Energy
from sys import argv, exit
from copy import copy
import json

def main():
    if len(argv) < 2:
        print("Usage: _'python3 _%s _<json-file>' _json-  
file _containing _data _about _the _system." %  
argv[0])
        exit(1)

    with open(argv[1], "r") as fp:
        obj = json.load(fp)

    sectors = {}
    energies = {}
    primaryenergies = {}
    energy = {}

    for energy in obj["primaryenergies"]:
        primaryenergies[energy] = Energy(obj["  
primaryenergies"][energy]["name"])

    energies = copy(primaryenergies)

    for sector in obj["sectors"]:
        pass
        sectors[sector] = Sector(obj["sectors"][sector  
]["name"])

        for energy in obj["sectors"][sector]["inputs"]  
]:
            energies[energy].add_input()

    print(energies)
    #print(sectors)

if __name__ == "__main__":
    main()
```

Och här är inputdatan till vårt program.

```
{
  "name": "Sveriges primärenergitillförsel",
  "primaryenergies": {
    "fossil": {
      "name": "Fossila bränslen"
    },
    "bio": {
      "name": "Biobränslen"
    },
    "nuclear": {
      "name": "Kärnkraft"
    },
    "water": {
      "name": "Vattenkraft"
    },
    "wind": {
      "name": "Vindkraft"
    }
  },
  "sectors": {
    "residences": {
      "name": "Bostäder",
      "inputs": {
        "electric": {
          "name": "El, specifik",
          "energy": 55,
          "efficiency": 100,
          "inputs": {
            "fossil": {
              "quota": 1.7,
              "efficiency": 35
            },
            "bio": {
              "quota": 3.8,
              "efficiency": 30
            },
            "wind": {
              "quota": 2.8,
              "efficiency": 100
            },
            "nuclear": {
              "quota": 39.7,
```



```

        "efficiency": 34
    },
    "water": {
        "quota": 47.8,
        "efficiency": 100
    },
    "spill": {
        "quota": 4.2,
        "efficiency": 100
    }
}
},
"heating": {
    "name": "Värme",
    "energy": 90,
    "efficiency": 100,
    "inputs": {
        "fossil": {
            "quota": 15,
            "efficiency": 80
        },
        "bio": {
            "quota": 13,
            "efficiency": 90
        }
    },
    "district_heating": {
        "name": "Fjärrvärme",
        "quota": 49,
        "efficiency": 90,
        "inputs": {
            "fossil": {
                "quota": 10,
                "efficiency": 75
            },
            "bio": {
                "quota": 61,
                "efficiency": 75
            }
        },
        "spill": {
            "name": "Spillvärme",
            "quota": 7,
            "efficiency": 100
        }
    },
    "pump": {

```

```

        "name": "Värmepump",
        "quota": 22,
        "efficiency": 250
    }
}
},
"el_heat": {
    "name": "El, uppvärmning",
    "quota": 23,
    "efficiency": 100
}
}
}
},
"transport": {
    "name": "Transport",
    "inputs": {
        "fuel": {
            "name": "Drivmedel",
            "energy": 18,
            "efficiency": 20,
            "inputs": {
                "fossil": {
                    "quota": 94,
                    "efficiency": 90
                },
                "bio": {
                    "quota": 6,
                    "efficiency": 50
                }
            }
        }
    }
}
},
"industry": {
    "name": "Industri",
    "inputs": {
        "electric": {
            "name": "El, specifik",
            "energy": "53",
            "efficiency": 100,
            "inputs": {
                "fossil": {

```

```

    "quota": 1.7,
    "efficiency": 35
  },
  "bio": {
    "quota": 3.8,
    "efficiency": 30
  },
  "wind": {
    "quota": 2.8,
    "efficiency": 100
  },
  "nuclear": {
    "quota": 39.7,
    "efficiency": 34
  },
  "water": {
    "quota": 47.8,
    "efficiency": 100
  },
  "spill": {
    "name": "Spillvärme",
    "quota": 4.2,
    "efficiency": 100
  }
}
},
"heating": {
  "name": "Värme",
  "energy": 121,
  "efficiency": 100,
  "inputs": {
    "fossil": {
      "quota": 56,
      "efficiency": 90
    },
    "bio": {
      "quota": 40,
      "efficiency": 90
    },
    "district_heating": {
      "name": "Fjärrvärme",
      "quota": 49,
      "efficiency": 90,
      "inputs": {

```

