

DYPENSI – Dynamic Pension Microsimulation Model for Slovenia: Overview, Model Architecture and Recent Developments towards a Multi-Purpose Policy Tool

www.eu-skladi.si

**Nataša Kump
Boris Majcen
Aleša Lotrič Dolinar
Joze Sambt
Martin Spielauer**

Outline

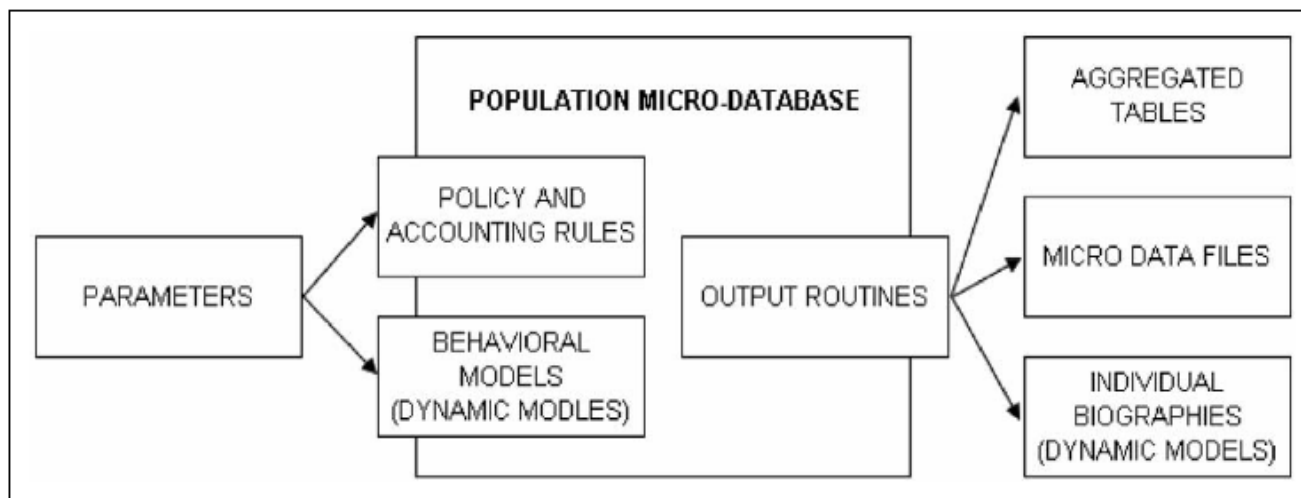
- Dynamic microsimulation
- Model description
- Model architecture
- Challenges
- Recent upgrades and planned extensions





Dynamic microsimulation (1)

- What is dynamic microsimulation?



Source: Spielauer, 2011. What is Social Science Microsimulation? Social Science Computer Review 29(1) 9-20

Dynamic microsimulation (2)



- When we should use dynamic microsimulation?
- Types of dynamic microsimulation models:
 - static vs. dynamic ageing
 - cohort model vs. population model
 - discrete vs. continuous time modelling
 - longitudinal vs. cross-sectional
 - open vs. closed model



Strengths and drawbacks

- Strengths from:
 - a theoretical perspective
 - a practical perspective
 - a technical perspective
- Drawbacks:
 - black box
 - model detail vs. prediction power: trade off between additional randomness and misspecification errors, between good aggregate predictions and a good prediction regarding distributional issues
 - development costs
- Some outcomes are aligned or calibrated toward aggregated numbers or projections obtained by external means

Previous DYPENSI release: basics (1)



- Model was initially developed in period 2011-2014
- The main goal of the model was to address the financial sustainability of the pension system.
- Modgen programming language (cross fertilization between models in Modgen).
- Time based (interacting population) continuous time model.
- Time of simulation: 2008 – 2060.
- Starting population: 113 thousands Slovenian residents living in 40 thousands households with socio-economic characteristics from year 2007 and wage and work-time histories.

Previous DYPENSI release: basics (2)



- Data adjustments (reweighting) to reflect 2007 population more realistically.
- Model is divided into three broad groups of modules: demographic, labour market, pension.
- Persons are linked in a simplified way with the (only) goal to model survivor pensions.
- Model is simultaneously aligned: demographic results are aligned to demographic projections and labour market states to labour force participation and unemployment rates projected by EC.
- Labour market events: first labour market entry, unemployment, re-employment, transition into and from activity.

DYPENSI updates and extensions



- Funding up to 2022
- Simulation period: 2018 to 2070
- New starting population from December 31, 2017
- Realistic family formation and family histories
- Updated DYPENSI will address pension adequacy besides the financial sustainability of the pension system → additional features needed
 - Social benefits and other incomes
 - Taxes and social contributions
 - Poverty rates estimations
- More realistic migration
- Retirement decisions

Model architecture

- Modgen programming language; previous release of DYPENSI has already been updated and adapted for use with Visual Studio 2017 and Modgen 12.1.
- High modularity:
 - demographic modules,
 - labour market modules,
 - pension and pension adequacy modules (other incomes, tax and social benefit policies child-related benefits, social assistance, income support, poverty rates and inequality indicators calculations)
- Alignment routines (optional): it enables us to reproduce external scenarios
- Time based continuous model → interacting population
- Two actors: person and state
- Results: output tables, microdata output at chosen time, histories of individuals



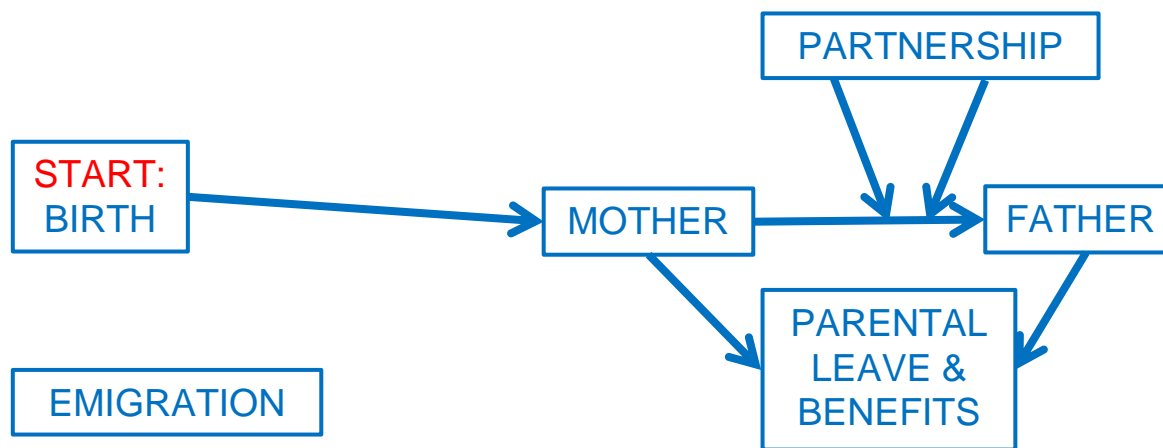


Treatment of time in DYPENSI

A continuous-time model allows for realistic sub-annual spell durations of processes:

- continuous-time events: births or search for parents, partnership formation (for females without partner at time of birth) and parental leave, first entry to the labour market, unemployment and unemployment wage compensation, deaths and survivor benefits, wage calculation
- monthly events (mid month or end month): alignment with labour market parameters, checking eligibility for survivor pension, old-age retirement, calculation of potential pension
- yearly events (mid year or end year): alignment of wages (optional), other incomes, update partnership status, emigration, disability pension, taxation, poverty rates

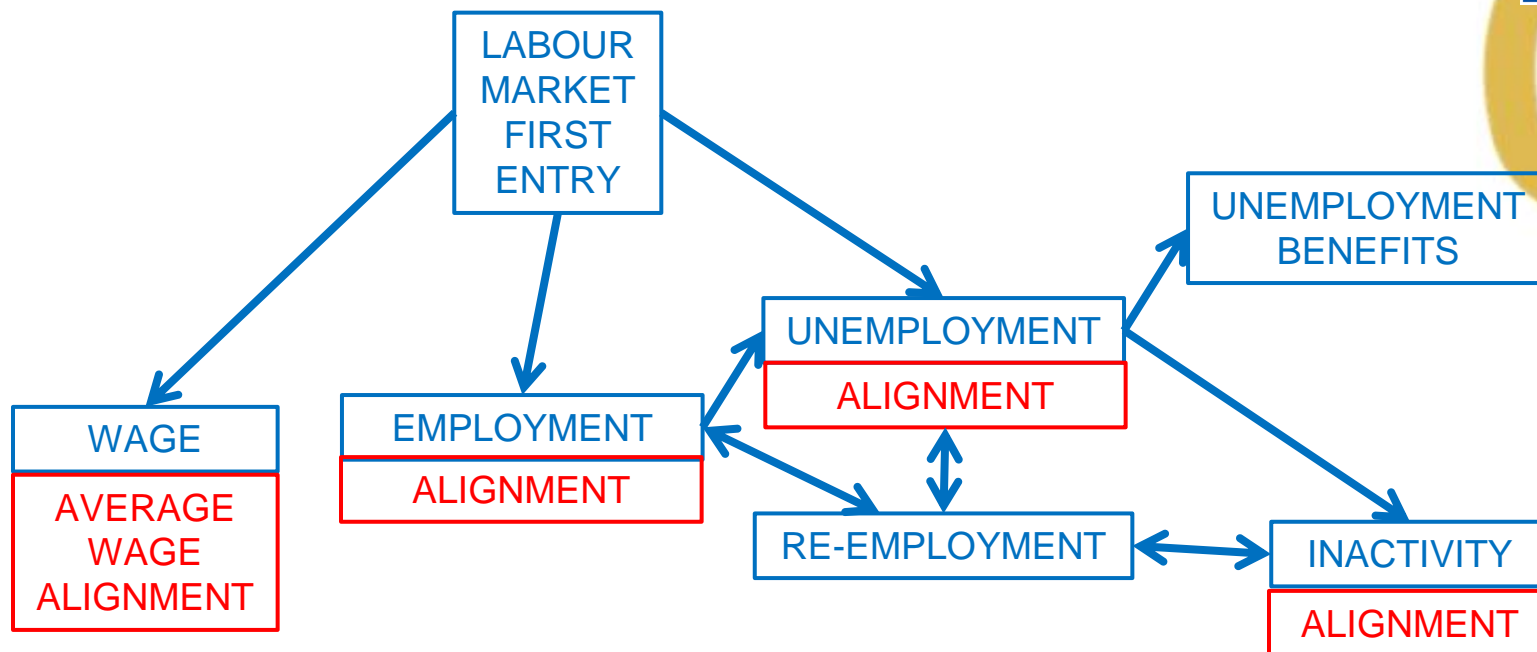
Model structure: births, migration and partnership



- There are three ways actors enter the simulation: starting population, future resident birth, immigration
- The number of births in the simulation is given by parameters; A mother is searched according to age- and parity- specific fertility rates.
- Partnership: formed at birth event. The female partnership status is updated at yearly events (in the middle of each year) according to observed partnership patterns by age, age of youngest child, and education.
- Immigration and emigration: the exact numbers of emigrates per age, sex and year as set in the parameter table are produced.

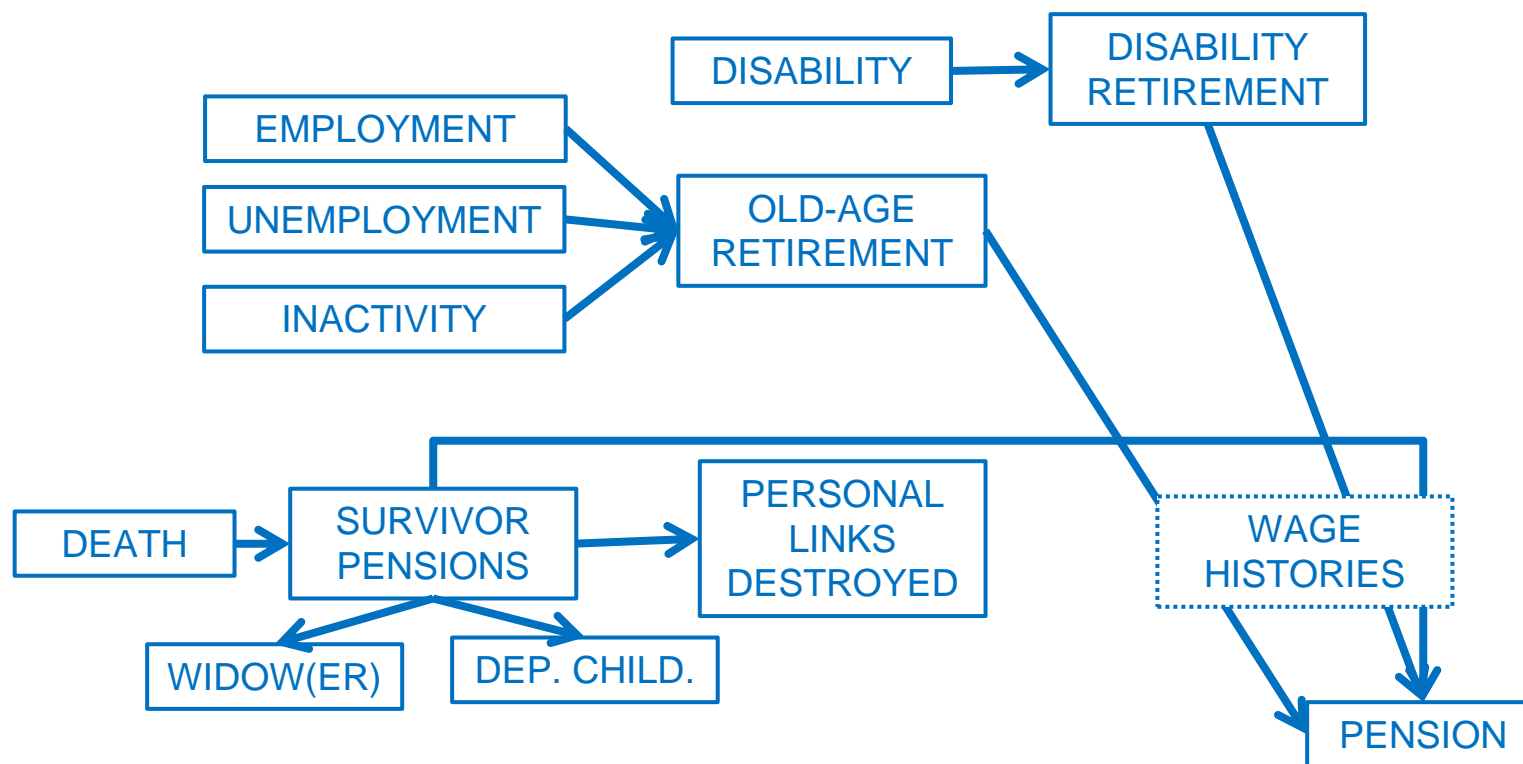


Model structure: labour market modules



- Labour force participation module: first entry, individual behaviour based on hazard rates, monthly alignment
- Employment module simulates changes from employment to unemployment and back: individual behaviour based on hazard rates, monthly alignment.

Model structure: retirement and pensions



- Old-age pension module: on a monthly basis, models pension eligibility and the potential pension amount at immediate retirement if eligible.
- The survivor pension module simulates survivor pensions at death event (children and widow-er). Recalculated each month.
- Disability event and disability pension: according to the probability to be disabled by age and sex (once per year).

Conclusions



- Work in progress
- Processes modelled more realistically (partnership formation and families, migration, retirement decisions, etc.)
- Development of DYPENSI to create a multipurpose policy tool covering various additional policy domains including social assistance, health, and long-term care

Thank you!