iMOPSE dataset instances generator - User's Guide

Marek E. Skowroński, Paweł B. Myszkowski

November 6, 2014

This document presents brief overview of iMOPSE dataset instances generator tool. The purpose of the tool is to generate new instances of MS–RCPSP.

Introduction

To launch this tool double–click on iGenerator.jar.

Technical requirements

To launch this tool, Java Runtime Environment (v.1.5 or newer) has to be provided.

General overview

After launching the application, the following screen is appeared.

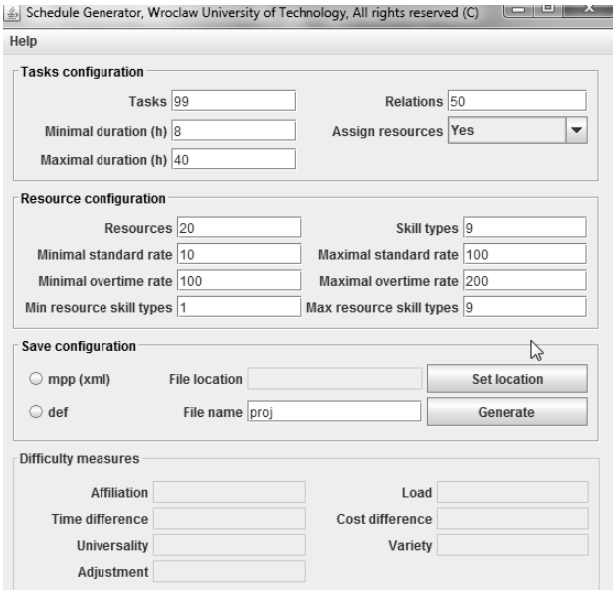

Figure 1: iMOPSE dataset instances generator – screenshot

Main sections could be then identified:

- Tasks & resources configuration – section with configuration settings for generating project elements – would be explained further in detail,
- Save configuration – section with file definition (name and location when the result should be saved on a disc),
- Difficulty measures – section representing difficulty measures for generated instance. For informative purposes only.

Main configuration

In this configuration the main elements of any project, according to MS–RCPSP, are defined: tasks, resources, precedence relations and skills definition.

Tasks

For simplicity, tasks are described by one main attribute: **duration**. It is set as a random value within the range between minimal and maximal duration. Range should be defined by positive values (could be float).

Note: tasks could be automatically assigned to any capable resources after ticking the checkbox *Assign resources*. The resulted schedule would be skill–feasible (preserving skills constraints).

Relations

Precedence relations are defined by one main attribute: **number of precedence relations** – defines how many precedence relations are in the project. This parameter does not involve the maximal number of precedence relations regarding to given resource. The density of precedence relations is not preserved in any way. Any task can have maximal number of precedence relations but can also have no precedence relations linked with.

Note: the general assumption has been made that only Finish–To–Start relations are defined.

Note #2: the number of precedence relations could be decreased slightly, because of trimming the number of tasks to match the upper bound of tasks number in a project. It is because of the recursive task generation process. Trimming tasks number could cause deleting number of precedence relations regarding tasks that are deleted.

Resources

Resource configuration is more parametrized than tasks and precedence relations configuration. The main elements are following:

- number of resources – defines how many resources would be defined in a project.
- salary rate
  - standard resource salary rate – defines the standard salary rate for resource.
  - overtime resource salary rate – defines the overtime salary rate for resource (not used in *def* file format).

  Both salary rates are set as a random value within the range of minimal and maximal potential values, set by User.

Skills

However skills are strongly connected with resources, this part of configuration would be described separately. The main elements of skills configuration regard following elements:

- number of skill types in a project
- number of skills owned by resource

There is no hierarchy between various skill types. Every type is totally independent from others.

Resource are defined by not only skill type but also a skill (familiarity) level. Constant, four values of skill level have been proposed, numbered from 0 (the lowest) to 3 (the highest). The higher the value is, the more proficient resource is in given skill.

Any resource can have only one skill with given type. Number of skills for resource is defined as a random value within the range between minimal and maximal potential skill types for resource, defined by User. Number of skills for resource should be smaller than number of skill types in a project.

Skills are also used to define skill required by given task to be performed. Resource can perform given task if it has required skill type in the familiarity level no lower than this task has. Skills for tasks are defined after setting skills for resource, to ensure that every task would have at least one resource that could be assigned and preserve the skill constraints.

Save configuration

In save configuration, three main elements should be set:

- output file format
- the directory where the resulted files would be stored
- the file name

If output file format would be set to *mpp (xml)* generator will generate *xml* file, according to the Microsoft Project (*mpp*) file format. Such a file could be straightforwardly loaded in Microsoft Project. If user would not set any of output file types or check *def*, the generator will generate a file in a simple *def* format that can be loaded in any text editor and can be used for scheduling algorithm testing without Microsoft Project.

If the directory would not be set, file would be saved in default directory (depending on the operating system). The name cannot be empty. Once the directory would be set, unlimited number of files could be saved there, until the generator is closed. Files with project definitions are stored in xml format that is ready to be opened in MS Project.

Warning: applying the same name for the file in given directory causes overwriting!

Difficulty measures

Difficulty measures presents the level of scheduling difficulty for given generated project instances. They are presented only for informative purposes.

Note: for given configuration, various difficulty measures could be obtained for newly generated instances, as some of proposed measures are related to non–deterministic features of this generator (e.g. duration / cost difference or adjustment).