Scrum Process Optimization Model

Gemini

September 5, 2025

Contents

1 Sets (Entities)

The model is based on the following sets, derived from the domain model entities. The index for each set is shown in parentheses.

- P: Set of all Projects (p)
- T: Set of all Teams (t)
- W: Set of all Workers (w)
- F: Set of all Features (f)
- S: Set of all Skills (s)
- R: Set of all Roles (r)
- PO: Set of all Product Owners (po)
- SM: Set of all Scrum Masters (sm)
- PB: Set of all Product Backlogs (pb)
- SP: Set of all Sprints (sp)
- SBL: Set of all Sprint Backlogs (sbl)
- SG: Set of all Sprint Goals (sg)
- E: Set of all Epics (e)
- US: Set of all User Stories (us)
- TSK: Set of all Tasks (tsk)
- BL: Set of all Blockers (bl)
- SH: Set of all Stakeholders (sh)
- VEL: Set of all Velocities (vel)
- REP: Set of all Release Plans (rep)

2 Indices

The following indices are used to iterate over the sets defined in Section 1.

- $p \in P$: Index for projects
- $t \in T$: Index for teams
- $w \in W$: Index for workers
- $f \in F$: Index for features
- $s \in S$: Index for skills
- $po \in PO$: Index for product owners
- $sm \in SM$: Index for scrum masters
- $sp \in SP$: Index for sprints
- $us \in US$: Index for user stories
- $tsk \in TSK$: Index for tasks
- $bl \in BL$: Index for blockers
- $rep \in REP$: Index for release plans

3 Goals

The objective function aims to optimize a weighted combination of several business goals. The overall model seeks to maximize the value Z.

$$\text{Maximize } Z = \sum_{i \in \text{Goals}} w_i \cdot G_i$$

The individual goals G_i are defined as follows:

- **G0: maximize_total_feature_priority**
  Maximize the business value by summing the priorities of completed features.

$$\max \sum_{f \in F} \text{priority}_f \cdot \text{select\_feature\_for\_release}_{f,rep}$$

- **G1: minimize_project_budget_overrun**
  This is represented as maximizing the negative cost to fit into the maximization function.

$$\max \left(- \sum_{p \in P} (\text{actual\_cost}_p - \text{budget}_p) \right)$$

- **G2: maximize_team_velocity**
  Maximize the average story points completed by teams.

$$\max \sum_{t \in T} \text{avg\_story\_points}_{vel_t}$$

- **G3: minimize_total_task_effort**
  Minimize the total effort spent on tasks for completed user stories.

$$\min \sum_{tsk \in TSK} \text{effort}_{tsk} \cdot \text{assign\_worker\_to\_task}_{w,tsk} \implies \max \left(- \sum_{tsk \in TSK} \text{effort}_{tsk} \cdot \text{assign\_worker\_to\_task}_{w,tsk} \right)$$

- **G7: maximize_story_points_per_sprint**
  Maximize the value delivered in each sprint.

$$\max \sum_{sp \in SP} \sum_{us \in US} \text{story\_points}_{us} \cdot \text{assign\_user\_story\_to\_sprint}_{us,sp}$$

4 Conditions

The optimization is subject to the following constraints, which ensure the model adheres to Scrum rules and project limitations.

- **C0: enforce_team_size_limit**
  The number of workers assigned to a team must be within a defined range [min_size, max_size].

$$\forall t \in T: \quad \text{min\_size} \leq \sum_{w \in W} \text{belongs\_to\_team}_{w,t} \leq \text{max\_size}$$

- **C2: project_budget_is_a_hard_cap**
  The sum of all costs associated with a project cannot exceed its budget.

$$\forall p \in P: \quad \sum \text{costs}_p \leq \text{budget}_p$$

- **C3: worker_must_have_skill_for_task**
  A worker can only be assigned to a task if they have the required skill. (Assuming a parameter $\text{RequiresSkill}_{tsk,s}$)

$$\forall w \in W, \forall tsk \in TSK, \forall s \in S: \quad \text{assign\_worker\_to\_task}_{w,tsk} \cdot \text{RequiresSkill}_{tsk,s} \leq \text{HasSkill}_{w,s}$$

- **C4: sprint_load_must_not_exceed_velocity**
  The total story points of user stories in a sprint cannot exceed the team's velocity.

$$\forall sp \in SP, \forall t \in T: \quad \sum_{us \in US} \text{story\_points}_{us} \cdot \text{assign\_user\_story\_to\_sprint}_{us,sp} \leq \text{velocity}_t$$

- **C7: task_assigned_to_one_worker**
  Each task must be assigned to exactly one worker.

$$\forall tsk \in TSK: \quad \sum_{w \in W} \text{assign\_worker\_to\_task}_{w,tsk} = 1$$

- **C9: worker_availability_is_a_constraint**
  The total effort of tasks assigned to a worker in a sprint cannot exceed their availability.

$$\forall w \in W, \forall sp \in SP: \quad \sum_{tsk \in \text{Tasks}_{sp}} \text{effort}_{tsk} \cdot \text{assign\_worker\_to\_task}_{w,tsk} \leq \text{availability}_{w,sp}$$

5 Decision Variables

The following variables are determined by the model to achieve the optimal solution.

- **DV0: assign_worker_to_task** ($x_{w,tsk}$)
  A binary variable indicating if worker w is assigned to task tsk.

$$x_{w,tsk} \in \{0,1\}, \quad \forall w \in W, \forall tsk \in TSK$$

- **DV1: assign_user_story_to_sprint** ($y_{us,sp}$)
  A binary variable indicating if user story us is assigned to sprint sp.

$$y_{us,sp} \in \{0,1\}, \quad \forall us \in US, \forall sp \in SP$$

- **DV2: assign_team_to_project** ($z_{t,p}$)
  A binary variable indicating if team t is assigned to project p.

$$z_{t,p} \in \{0,1\}, \quad \forall t \in T, \forall p \in P$$

- **DV3: select_feature_for_release** ($a_{f,rep}$)
  A binary variable indicating if feature f is included in release plan rep.

$$a_{f,rep} \in \{0,1\}, \quad \forall f \in F, \forall rep \in REP$$

- **DV5: set_task_effort** (e_{tsk})
  An integer variable for the effort of task tsk.

$$e_{tsk} \in Z^+, \quad \forall tsk \in TSK$$