Mathematical Model for SCRUM-Based Software Development Optimization

AI Assistant

September 4, 2025

Contents

Introduction

This document provides a formal mathematical formulation of an optimization model for a software development company utilizing the SCRUM framework. The model is derived from a defined domain of entities, relationships, goals, conditions, and decision variables. The objective is to optimize resource allocation, scheduling, and feature selection to maximize efficiency and value delivery while minimizing cost and delays under a set of constraints.

1 Sets (Entities)

- Project = $\{p_1, p_2, ..., p_{|\mathrm{P}|}\}$: Set of all projects.
- Team = $\{t_1, t_2, ..., t_{|\mathrm{T}|}\}$: Set of all teams.
- Worker = $\{w_1, w_2, ..., w_{|\mathrm{W}|}\}$: Set of all workers/employees.
- Feature = $\{f_1, f_2, ..., f_{|\mathrm{F}|}\}$: Set of all features.
- Skill = $\{s_1, s_2, ..., s_{|\mathrm{S}|}\}$: Set of all skills.
- Role = $\{r_1, r_2, ..., r_{|\mathrm{R}|}\}$: Set of all roles.
- ProductOwner = $\{po_1, po_2, ..., po_{|\mathrm{PO}|}\}$: Set of Product Owners.
- ScrumMaster = $\{sm_1, sm_2, ..., sm_{|\mathrm{SM}|}\}$: Set of Scrum Masters.
- ProductBacklog = $\{pb_1, pb_2, ..., pb_{|\mathrm{PB}|}\}$: Set of Product Backlogs.
- Sprint = $\{sp_1, sp_2, ..., sp_{|\mathrm{SP}|}\}$: Set of sprints.
- SprintBacklog = $\{sbl_1, sbl_2, ..., sbl_{|\mathrm{SBL}|}\}$: Set of Sprint Backlogs.
- UserStory = $\{us_1, us_2, ..., us_{|\mathrm{US}|}\}$: Set of user stories.
- Task = $\{tsk_1, tsk_2, ..., tsk_{|\mathrm{TSK}|}\}$: Set of tasks.
- Blocker = $\{bl_1, bl_2, ..., bl_{|\mathrm{BL}|}\}$: Set of blockers.
- Stakeholder = $\{sh_1, sh_2, ..., sh_{|\mathrm{SH}|}\}$: Set of stakeholders.
- ReleasePlan = $\{rep_1, rep_2, ..., rep_{|\mathrm{REP}|}\}$: Set of release plans.

2 Indices

- $p \in$ Project
- $t \in$ Team
- $w \in$ Worker
- $f \in$ Feature
- $s \in$ Skill
- $sp \in$ Sprint
- $sbl \in$ SprintBacklog
- $us \in$ UserStory

- $tsk \in$ Task
- $bl \in$ Blocker
- $sh \in$ Stakeholder
- $rep \in$ ReleasePlan

3 Goals

- **G0: minimize_total_project_cost** - Minimize the total budget consumed across all projects.

$$\text{Minimize } Z_0 = \sum_{p \in \text{Project}} \text{budget}(p)$$

- **G1: maximize_team_velocity** - Maximize the average amount of work a team can complete per sprint.

$$\text{Maximize } Z_1 = \sum_{t \in \text{Team}} \text{velocity}(t)$$

- **G2: minimize_blocker_resolution_time** - Minimize the average time between a blocker being detected and resolved.

$$\text{Minimize } Z_2 = \sum_{bl \in \text{Blocker}} (\text{resolved\_on}(bl) - \text{detected\_on}(bl))$$

- **G3: maximize_feature_priority_score** - Maximize the total sum of priority scores for features selected in a release.

$$\text{Maximize } Z_3 = \sum_{f \in \text{Feature}} \text{priority}(f) \cdot x_f \quad \text{where } x_f \in \{0, 1\} \text{ indicates selection}$$

- **G4: minimize_sprint_planning_duration** - Minimize the total time spent in sprint planning meetings.

$$\text{Minimize } Z_4 = \sum_{spp \in \text{SprintPlanning}} \text{duration}(spp)$$

- **G5: maximize_team_availability** - Maximize the cumulative availability of all team members.

$$\text{Maximize } Z_5 = \sum_{w \in \text{Worker}} \text{availability}(w)$$

- **G6: minimize_task_effort_variance** - Minimize the difference between estimated and actual effort for tasks.

$$\text{Minimize } Z_6 = \sum_{tsk \in \text{Task}} |\text{estimated\_effort}(tsk) - \text{actual\_effort}(tsk)|$$

- **G7: maximize_stakeholder_satisfaction** - Maximize the average influence level of satisfied stakeholders.

$$\text{Maximize } Z_7 = \sum_{sh \in \text{Stakeholder}} \text{influence\_level}(sh) \cdot y_{sh} \quad \text{where } y_{sh} \in \{0, 1\} \text{ indicates satisfaction}$$

- **G8: minimize_technical_debt** - Minimize the number of features or stories with a status indicating incomplete technical quality.

$$\text{Minimize } Z_8 = \sum_{f \in \text{Feature}} \mathbb{I}(\text{status}(f) = \text{'Technical Debt'})$$

- **G9: maximize_sprint_goal_achievement** - Maximize the number of sprints where the achievement status is marked as complete.

$$\text{Maximize } Z_9 = \sum_{sg \in \text{SprintGoal}} \mathbb{I}(\text{achievement\_status}(sg) = \text{'Complete'})$$

4 Conditions

- **C0: team_size_constraint** - The size of a team must be within a specified range (e.g., 5-9 members).

$$5 \leq \text{team\_size}(t) \leq 9 \quad \forall t \in \text{Team}$$

- **C1: project_deadline_must_be_met** - The project end date must not exceed a fixed deadline D_p.

$$\text{project\_end}(p) \leq D_p \quad \forall p \in \text{Project}$$

- **C2: sprint_duration_fixed** - A sprint must have a fixed duration D_{sprint}.

$$\text{end\_date}(sp) - \text{start\_date}(sp) = D_{sprint} \quad \forall sp \in \text{Sprint}$$

- **C3: worker_availability_limit** - An individual worker's availability cannot exceed 100%.

$$0 \leq \text{availability}(w) \leq 1 \quad \forall w \in \text{Worker}$$

- **C4: budget_not_exceeded** - The total cost of all projects must not exceed the overall organizational budget B_{org}.

$$\sum_{p \in \text{Project}} \text{budget}(p) \leq B_{org}$$

- **C5: feature_requires_skills** - A feature can only be assigned to a team if the team's collective skills meet the feature's requirements. Let $\text{req\_skills}(f)$ be the skills required for feature f and $\text{team\_skills}(t)$ be the skills possessed by team t.

$$\text{req\_skills}(f) \subseteq \text{team\_skills}(t) \quad \forall f \text{ assigned to } t$$

- **C6: unique_worker_assignment** - A worker can only be assigned to one primary team at a time.

$$\sum_{t \in \text{Team}} \text{assign\_worker\_to\_team}(w, t) \leq 1 \quad \forall w \in \text{Worker}$$

- **C7: sprint_scope_capacity** - The total story points in a sprint backlog cannot exceed the team's known velocity V_t.

$$\sum_{us \in \text{SprintBacklog}} \text{story\_points}(us) \leq V_t \quad \forall \text{ sprint backlogs } sbl \text{ for team } t$$

- **C8: blocker_severity_requires_attention** - Blockers with 'Critical' severity must be resolved within 24 hours.

$$(\text{resolved\_on}(bl) - \text{detected\_on}(bl)) \leq 24 \quad \forall bl \in \text{Blocker} \mid \text{severity}(bl) = \text{'Critical'}$$

- **C9: definition_of_done_met** - A user story cannot be marked 'Done' unless all acceptance criteria are met. Let ac_met(us) be the proportion of acceptance criteria met.

$$\text{status}(us) = \text{'Done'} \implies \text{ac\_met}(us) = 1 \quad \forall us \in \text{UserStory}$$

- **C10: product_owner_approval_required** - A feature cannot be included in a release plan without Product Owner approval. Let approved(f) be a binary indicator.

$$x_f \leq \text{approved}(f) \quad \forall f \in \text{Feature}, \forall rep \in \text{ReleasePlan}$$

5 Decision Variables

- assign_worker_to_team$_{w,t} \in \{0, 1\}$: Binary assignment of worker w to team t.
- select_feature_for_release$_{f,rep} \in \{0, 1\}$: Binary decision to include feature f in release plan rep.
- allocate_sprint_budget$_{sp} \in [0, 100000]$: Continuous budget allocation for sprint sp.
- set_team_size$_t \in \{3, 4, ..., 12\}$: Integer variable for the size of team t.
- assign_story_points$_{us} \in \{1, 2, ..., 21\}$: Integer story points for user story us.
- set_feature_priority$_f \in \{1, 2, 3, 4, 5\}$: Integer priority level for feature f.
- plan_sprint_duration$_{sp} \in \{7, 8, ..., 30\}$: Integer duration (days) for sprint sp.
- assign_task_effort$_{tsk} \in [0.5, 40]$: Continuous effort estimate (person-hours) for task tsk.
- set_worker_availability$_w \in [0, 1]$: Continuous availability percentage for worker w.
- blocker_resolution_resource$_{bl} \in \{0, 1, ..., 100\}$: Integer resource allocation for blocker bl.
- select_stakeholder_for_review$_{sh,sr} \in \{0, 1\}$: Binary invitation for stakeholder sh to review sr.
- assign_skill_to_worker$_{w,s} \in \{0, 1\}$: Binary certification of skill s for worker w.