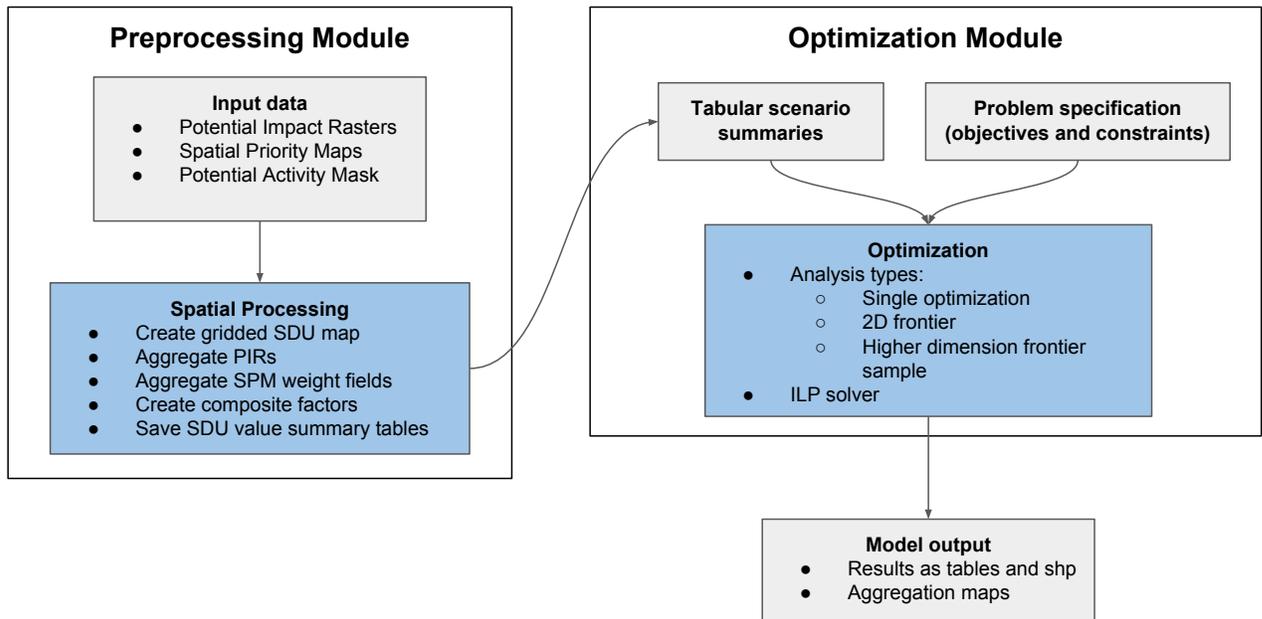


## Introduction

Spatial planning is a complex process that must confront a myriad of informational and deliberative challenges. The Restoration Opportunities Optimization Tool (ROOT) helps address these challenges as they apply to the generation and interpretation of ecosystem service information in planning processes.

A ROOT analysis identifies areas with the highest potential to increase ecosystem service (ES) provision under alternative management scenarios using linear programming optimization. Based on user-generated analyses of potential ES changes across potential “activity” sites, mapping of key service endpoints, and decision-specific constraints, ROOT calculates optimal portfolios of restoration locations to support multiple ecosystem service objectives. The primary outputs from ROOT analyses are a suite of optimized restoration allocations based on different prioritizations of each ES objective, a synthesis map that combines these allocations, and various tables summarizing potential benefits for each allocation decision.

Prior to running ROOT, the user needs to conduct a preliminary ecosystem services analysis. This entails running InVEST or other ES models to estimate the potential impact of some restoration or management change. When the ES model pertains to a biophysical process without including valuation, it will often be necessary to connect the analysis to an impacted endpoint to measure the expected change in a service. To include this in ROOT, the user’s analysis should create maps that assign priority scores to areas landscape based on where change in the physical process will be more or less valuable.



**Figure 1:** ROOT workflow. The program consists of two main modules that can be used independently, but make up the full processing flow. The preprocessing module converts spatial raster and vector data to tabular scenario summaries. The optimization model uses these, along with an optimization problem specification to drive an ILP solver to generate optimal landscapes. Once the preprocessing step has been run, the optimization may be repeated with multiple problems, or the addition of new tabular data.

## Data file descriptions

### Activity mask raster (AMR)

This raster identifies which pixels are considered as potential sites for the activity to take place. This allows the apply to use relevant screening criteria to restrict the analysis to likely or suitable locations. When ROOT processes the inputs, it will calculate the total benefits for each ecosystem service, the overlap of these benefits with areas important for the delivery of ecosystem services (i.e. servicesheds), and an activity area only for pixels included in the activity mask. An appropriate mask should exclude all pixels where an activity is not feasible, but should not exclude pixels based on expected benefits or costs, as these factors will be considered separately by the optimization routine, and in some cases, areas that initially seem too costly or not good enough will turn out to be good choices based on other considerations in the optimization.

### Impact Potential Map

The first type of data input is the impact potential map (IPM). These rasters represent the expected change in a given process or service occurring in each pixel if the activity is implemented. In many cases, the IPM will directly calculate the difference between the activity

value ( $V_{\text{activity}}$ ) which represents the value of ecosystem services based on a future or restored scenario and the baseline value ( $V_{\text{baseline}}$ ) which is the current value of an ecosystem service:

$$V_{\text{marginal}} = V_{\text{activity}} - V_{\text{baseline}}$$

If these values are calculated using InVEST, two runs of each model are required. First, a baseline run with a LULC map representing the current landscape, and second, an activity run with all pixels in the activity mask converted to the appropriate post-activity LULC or appropriate changes to biophysical tables if the proposed activities change the qualities of a LULC class but do not change the categorization itself. The IPM is calculated as the difference between the outputs of these two scenarios. In other cases, other methods could be used to generate an index map, or some other estimate of potential impact.

### Spatial weighting maps (SWM)

Spatial weighting maps (SWMs) represent additional factors that modify the importance of a given impact potential map for different areas of the landscape. For example, land areas with runoff that flows into hydropower dams' reservoirs may be higher priority areas for reducing sediment export, or changes in agricultural productivity may have more weight in areas with higher rural poverty. SWMs allow the user to apply multiplicative weighting factors to the IPMs to create a priority- or value-adjusted measure of ES change in each potential activity allocation area. Spatial weights should be encoded as shapefiles, with geometries that outline areas with different prioritizations, and fields that contain the priority weights.

### Servicesheds

One common reason to use SWMs is to identify and prioritize the areas on a landscape that affect service-demanding locations. Analogous to the concept of a watershed, which outlines the area of landscape that drains through a particular point, a serviceshed outlines the area where changes to a particular biophysical process will affect people, infrastructure, or some other prioritized endpoint. A serviceshed SWM could be a simple 0 or 1 binary map, or capture differential potential impact across the identified range or differences in priority between different servicesheds. For example, rates of rural poverty could be used as a weighting factor in a SWM that delineates areas of high agricultural productivity

### Composite factors

ROOT requires the user to specify how the MVRs and SWMs should be combined; that is, which spatial weights correspond to which marginal values. The composite factor table (CFT) allows the user to create any number of combined values by listing the MVRs and fields in SWM shapefiles, which will then be multiplied together and saved as part of the initial preprocessing steps. These composite factors can be used as the objectives to be maximized or minimized in the optimization, or be used to define constraints.

### Spatial Decision Unit (SDU) map

The spatial decision unit map is either provided by the user or created by ROOT to aggregate the other spatial data to a consistent size and reduce the number of spatial units to a size amenable to optimization. Automatic spatial decision unit maps will be made up of squares or

hexagons, depending on user input, with a constant area, also specified by the user. In the optimizations, ROOT will determine which SDUs should be chosen for implementation of the restoration activity, with the assumption that all potential activity pixels within an SDU are either chosen or not (except for the “last” SDU chosen to fill a constraint, which will have a fractional allocation).

## What the tool does

Tool operation is separated into two phases, preprocessing and optimization. A workflow may consist of one preprocessing step followed by multiple optimization analyses. Different optimizations may be used to evaluate the effect of different constraint values, inclusion of different weighting factors, or other variations on the optimization problem to be solved. The separation between the steps is primarily to save time, as the preprocessing operations may be identical across optimizations in the same context.

### Preprocessing

This phase converts the spatial inputs (MVRs and SWMs) to tabular form for use by the optimizer.

1. **Create Spatial Decision Unit (SDU) grid.** The first step is to create a grid of regular polygons covering the activity mask pixels. Based on user inputs, the tool will create an equal-area square or hexagon-based grid. The tool will exclude areas of the landscape containing no potential activity pixels. Any area containing even a single pixel in the activity mask will be included, so preprocessing to remove isolated or very small clusters might be appropriate.
2. **Aggregate Marginal Value Rasters.** The tool calculates the sum of the MVR pixels contained in each SDU polygon. Only the values of pixels that overlap with the AMR are counted towards the SDU total.
3. **Aggregate Serviced Maps.** For each SDU, the tool calculates the fraction of the grid polygon overlapping each SWM, and the average value (weighted by the area of each intersection between the polygon in the SWM and the SDU grid cell) of each indicated prioritization field. The fields to summarize are specified by the user in an input table to the tool (see below)
4. **Save results.** The results are saved to tables called ‘baseline.csv’ and ‘marginals.csv’. A column called ‘maskpixha’ (mask pixel hectares) is added indicating the total area of potential activity pixels per SDU. The results are also joined to a shapefile called ‘sdu\_grid.shp’. For full details on the columns included in this table, see below.

### Optimization

The exact operation of the optimization phase depends on the mode selected by the user. In general, the tool will run a number of optimization analyses corresponding with different potential prioritizations of each objective. These results are saved as individual solution tables, a summary table, and an agreement map.

- Solution tables: These tables, called 'solution\_X.csv', describe the results of a single optimization run. The column 'solution\_value' gives the proportion of each SDU chosen for activity implementation.
- Summary table: The summary table sums the objective scores for all optimization runs. This allows the user to investigate how solutions differ in the net changes to objectives and create tradeoff curves or other multi-objective comparison figures or analyses.
- Agreement map: The agreement map provides a spatial summary of the optimization runs. The file 'agreement.shp' has a field 'agreement' that indicates, for each SDU, the number of optimization runs that selected that SDU for activity implementation. The user can identify SDUs that are selected by nearly all optimizations, which indicates they are high priority regardless of objective weighting, as well as SDUs that are rarely or never selected, which are low priority across all weights. SDUs with intermediate selection rates represent the critical decision areas where prioritization between objectives may determine the final choice during implementation of proposed activities within a landscape.

Depending on the number of objectives or analysis interest, there are several optimization modes (selected by entering the name of the mode in the field "Optimization type" in the tool's interface).

- single: For the case that specific weights on each objective are known ahead of time, this mode allows the user to run a single optimization. The tool will multiply the objective columns by the indicated weights, and then run the optimization.
- frontier: **2 objectives only**. This mode creates a tradeoff curve between two named objectives. It ensures that the solutions fall along all portions of the tradeoff, so is well suited to producing tradeoff graphics.
- n\_dim\_frontier: **2 or more objectives**. This is the default when more than two objectives are important to the analysis. It will perform optimizations using randomly generated weights to obtain a sample of the 2+ dimensional tradeoff curve.
- n\_dim\_outline: This mode allows the user to be sure that they've captured the full extent of a multi-dimensional tradeoff. It runs the "frontier" analysis for all 2-objective pairs from a set of 3 or more objectives.

### Optimization problem

Formally, the tool solves a linear program with area per SDU  $i$ ,  $x_i$ , as the decision variable. For each optimization, the objective function is a weighted sum of columns selected from the preprocessing output tables, baseline.csv and marginals.csv. These columns are typically serviceshed-weighted biophysical marginal values.

$$\max_{x_i} \sum_s \sum_i w_s F_{si} x_i$$

Where

$x_i$ : Decision variable, with value between 0 and 1, indicating fraction of potential the potential activity area in SDU  $i$  to choose for implementation.

$w_s$ : Weight assigned to service (objective)  $s$  for this run of the optimizer

$F_{si}$ : value of the (composite) factor  $F$  for service  $s$  in SDU  $i$ .

The final objective is thus the sum of all benefits across the landscape, weighted by optimization run variables.

In practice, not all objectives will have a specific serviceshed weighting or marginal value raster. For example, the value of carbon sequestration may not depend on location given its global, rather than local, benefits, or the user may have added a monetary cost column to the baseline and marginal tables after preprocessing. In these cases, the equation above doesn't apply directly, but the weighted sum objective is the same.

The tool also allows the user to specify constraints. These represent requirements that a solution must meet to be considered satisfactory. Some examples are total activity area equaling a target area, cost fitting a given budget, or minimum values for improvement for a particular service.

## Running the tool

This section explains the fields in the user interface and the various files that must be provided to perform a run of the model.

**Workspace:** The folder on the user's hard drive that will be used to save intermediate and final results.

**Results Suffix:** optional argument that can be used to create separate output files for different runs.

## Preprocessing Section

### Data preparation notes

The tool requires that all spatial data (rasters and vectors) have the same projection, and additionally, that all rasters (the mask and each impact potential raster) have identical extent and cell size. The tool will quit with an error message if these conditions aren't met.

### Interface fields

- **Do preprocessing:** Checkbox to enable or disable preprocessing. When enabled, ROOT will construct the tabular data needed to perform the optimization. If users have already run the preprocessing once and just want to change the optimization parameters (analysis type, number of frontier points, objectives, or targets), they can uncheck this box to skip repeating the spatial pre-processing.
- **Impact Potential Raster Table (IPRT):** File path to a .csv file on the user's computer that contains a list of the impact potential rasters and the location of these raster files on the user's computer to use in the optimizations. The file should have columns "name", and "file\_path". The entries in "name" should be user-generated short names for the impact potential map, with no spaces. These names will be used later to refer to the impact potential data to construct combined factors and in the optimization step, so should be descriptive, but are completely up to the user.

name	file_path
n_export	path (e.g. D:/users/BeattyC/ROOT/MarginalRasters/MVnitrogen.tif)
s_export	path
cost	path

The values from these rasters summarized to the SDU scale will be added to the results table as in a column with the label given in the “name” column in the IPRT.

- Spatial Weight Maps Table (SWMT):** File path to a .csv file on user’s computer that contains a list of the shapefiles that define servicedshed or other spatial weighting factors. The file should have columns “name”, “file\_path”, and “weight\_col”. “name” and “file\_path” are the same as the corresponding columns from the IPRT, while “weight\_col” specifies which column(s) to pull from the shapefile as weighting attributes. To extract more than one column from the same shapefile, separate the field names with a space (note this means that the field names themselves cannot contain spaces).

name	file_path	weight_col
hydropower	path	capacity depth
surfacewater	path	pop impair

The summarized values from the SWMs will be added to the results table in two ways. The fraction of each SDU overlapping polygons in the SWM will be given the label from the “name” column in the SWMT. The weighted averages for each of the fields listed in “weight\_col” will be given labels combining the “name” and “weight\_col” values by joining them with an underscore (‘\_’). For example, in the above example, three columns will be created for the hydropower SWM, ‘hydropower’, ‘hydropower\_capacity’, and ‘hydropower\_depth’, and similarly for the surfacewater SWM.

- Composite Factors Table:** A .csv file that allows the user to combine named factors from the impact potential rasters or spatial weighting shapefiles to construct additional composite factors. This is the primary way to apply servicedshed weighting to a biophysical process to optimize for the resulting ecosystem service. The table has two columns, “name”, which allows the user to assign a short name to the factor to be created, and “factors”, the list of impact potential values or spatial weighting values to be multiplied together. The list of factors to combine specified in the “factors” field must be space-separated. The names of factors generated from servicedshed shapefiles are the name of the shapefile (from the ‘name’ column) and the field name joined with an underscore (‘\_’). The name of the shapefile alone can be used to refer to the fractional overlap between the servicedshed polygons and the SDU.

name	factors
sed_hyd	s_export hydropower_capacity

sed_sw	s_export surfacewater_impair surfacewater_pop
n_sw	n_export surfacewater_impair surfacewater_pop

This example table will create three new values for each SDU in the output table, representing sediment impacts on hydropower and surface water for drinking, and nitrogen impacts on surface water for drinking. The hydropower score is created by multiplying the export from each SDU with the capacity of the downstream reservoir. The drinking water scores multiply the export with an impairment index and the number of people extracting water downstream from the SDU.

- **Potential Conversion Mask:** Path to a GeoTIFF file on user’s computer. This raster specifies which pixels should be aggregated to the spatial decision unit polygons in the preprocessing step, and which should be excluded. Pixels must have values of 1 or NODATA, respectively.
- **Spatial Decision Unit Shape:** This field specifies the shapes that will be used to aggregate the Impact Potential Rasters and Spatial Weight Maps. If the user already has a specific SDU shapefile, the full file path to the .shp file should be entered here. This shapefile will be copied to the workspace as *workspace/sdu\_grid.shp*. The provided shapefile must contain a field SDU\_ID with unique ID numbers for each SDU polygon. Alternatively, ROOT can generate a grid of squares or hexagons instead. In this case, the user should enter either “square” or “hexagon”. A grid with the appropriate size (see next field) will be created as *workspace/sdu\_grid.shp*.
- **Spatial Decision Unit Area:** This value is used to determine the size (area) of the regular grid units if “square” or “hexagon” is entered in the previous field. It is ignored if a shapefile path is provided.

### Optimization Fields

- **Analysis type:** One of “single”, “frontier”, “n\_dim\_frontier”, “n\_dim\_outline”.
  - single: solves a single optimization with specific weights for each objective.
  - frontier: 2D frontier, ensures even spacing of points along a tradeoff curve for improved visualization.
  - n\_dim\_frontier: multi-objective, uses randomly selected weights to sample from the higher-dimensional tradeoff curve. The random generation process is designed to evenly sample the portion of a sphere centered at (0, 0, 0) with all non-negative weights.
  - n\_dim\_outline: set of all 2-dimensional frontier pairs given 3 or more objectives.
- **Number of frontier points:** Number of solutions to find.
- **Objectives Table:** This table identifies the factors to optimize for, and whether they should be maximized or minimized. Fields in the “name” column can refer to any of the factors named in the IPRT, SWMT, or CFT. The data that will be used by the optimizer is found in *workspace/marginals/marginal.csv*. Any numeric column from that table may be used. Fields in the “weight” column will be either numeric values (weights) for the “single” analysis type, or “min” or “max” (or “minimum”, “minimize”, etc...) for the other analysis types.

name	weight
sed_hyd	min
carb_seq	max
cost	min

In this example, the optimizer solutions will maximize carbon sequestration (carb\_seq) and minimize sediment export in catchments of hydropower reservoirs (sed\_hyd) and the cost of the activity (cost). The exact priority given to each objective will depend on the weights for each particular point on the frontier.

- **Targets Table:** Path to a csv file on user’s computer. This csv file describes targets (aka constraints) to impose on the optimization solutions. It should have columns “name”, “cons\_type”, and “value”.
  - **name:** the name of one of the impact potential or composite factors. Note that the automatically generated column “maskpixha” can also be used to set a constraint on the total area to restore. It specifies the area (in hectares) of potential conversion pixels contained in each SDU polygon.
  - **cons\_type:** ‘=’, ‘<=’, or ‘>=’. Specifies the type of constraint to impose. The summed value for **name** will have this relationship to **value**.
  - **value:** the numeric value for the target

name	cons_type	value
maskpixha	=	20000
nit_red	>=	1000

Example: in this example table, the solutions will have restored area equal to 20,000 hectares and change in nitrogen will be a reduction of at least 1000 kg.

## Outputs from ROOT

This section describes the files generated by the preprocessing and optimization steps.

### Preprocessing

The preprocessing step creates two primary outputs

1. sdu\_grid.shp. This is a shapefile containing regular squares or hexagons covering the area of interest. It excludes areas containing no values in the mask raster.
2. marginals.csv. This table is found in the marginal\_tables folder in the workspace and summarizes the values of the raster, shapefile, and combined factors by SDU. Columns in this table are:
  - SDU\_ID: identification number for the SDU. Matches the same column in the attribute table of sdu\_grid.shp.
  - pixel\_count: total number of pixels inside the SDU polygon
  - area\_ha: area of the pixels inside the SDU polygon

maskpixct: number of pixels in the SDU polygon with a value of 1 in the activity mask raster

maskpixha: area in hectares of these pixels

factors: the rest of the columns are the aggregated values from the impact potential rasters, spatial weighting maps, and the composite factors created in the preprocessing step.

- Note for advanced users: it is possible to add additional columns to this file after preprocessing, and these columns can be referred to in the optimization step. Note that an identically-named column must also be created in baseline.csv with appropriate values.

## Optimization

Outputs from the optimization consist of individual solution files, a summary table, and agreement map.

1. Summary table: the summary table provides full-landscape summary statistics for each of the solutions created during the optimization step. It sums the values of all factors in SDUs that were selected for activity implementation. These summary values can be used to plot the tradeoff curve (or efficiency frontier) for the solutions.
2. Individual solution files: A set of tables named optimization\_solutionsX.csv, which give the SDU-specific decisions corresponding with the points summarized in the summary table. Particular solutions can be visualized by joining these tables with sdu\_grid.shp and choosing 'solution\_value' as the visualization field.
3. Agreement map: This shapefile provides a summary of the set of optimizations. The field 'choice\_ct' counts the number of times each SDU was chosen for activity implementation in the set of optimization runs. SDUs where the choice count is very high or low have strong agreement across all points in the analysis. These SDUs are either nearly always or nearly never chosen, regardless of the weights assigned to the objectives. On the other hand, SDUs with an intermediate choice count are only chosen for particular combinations of weights.
4. weight\_table: for the n\_dim\_frontier analysis, a table is created that lists the weighting values used in each optimization.

## Tips and troubleshooting

- **Getting file paths:** The marginal value raster table and servicedshed shapefile table require you to enter the path (location) of each file into the table. There are hidden but easy ways to do this on Windows and Mac:
  - **Windows:** Hold shift and right click the file. Select "Copy as path" in the menu that appears. You can then paste this directly into the csv file.
  - **Mac:** Right click the file, then press and hold the option key. Select "Copy \_ as Pathname", and the text is available to paste into the csv file.
- **Spatial data preparation:**

- Be sure that rasters and shapefiles have identical projections, and that all rasters have the same extent and cell size. ROOT will fail if these conditions are not met.
- Make sure that the Potential Conversion Mask has only 1 or NODATA values, where the activity is permitted or not, respectively.
- **Number of SDUs:** Analyses with a large number of SDUs may take an extremely long time to complete or run out of memory on the user's computer. We suggest starting with a smaller number of large SDUs, and iteratively running with smaller and smaller SDUs to ensure that solutions can be computed at each step. A good starting point for fast analysis is ~1000 SDUs (though this can depend heavily on the number of constraints involved and the user's computer).
- **Similarly-sized objectives:** Numerical optimizers, like the one this tool uses, are more likely to find solutions if the objectives (and constraints) are of similar orders of magnitude. If you have some variables that are many times larger or smaller than others and the solver is failing, consider rescaling them so they have similar sizes. This is particularly likely to be an issue for the "single" optimizer, which does not do any internal rescaling.

## Contacts:

**NatCap Lead Designer and Developer:** Peter Hawthorne (phawthorne@umn.edu)

**NatCap Designer:** Adrian Vogl (avogl@stanford.edu)

**IUCN Designer:** Craig Beatty (Craig.BEATTY@iucn.org)

*ROOT was developed by NatCap in partnership with the International Union for the Conservation of Nature (IUCN), with support from UK Aid.*