Open Risk

Solstice Manual:: Analytic Framework

Version: 0.1

Release Date: November 24, 2022

www.openriskmanagement.com

The open future of risk management

Open Risk
Solstice Manual:: Analytic Framework

About Open Risk

Open Risk is an independent provider of training and risk analysis tools to the broader financial services community. Our mission is captured by the motto: The open future of risk management. Learn more about us at: www.openriskmanagement.com and join the discussion at: www.openriskcommons.org

Further Resources

The Open Risk Academy offers a range of online courses around sustainable finance, risk and portfolio management, which utilize the latest in interactive eLearning tools. Please inquire at **info@openrisk.eu** about course schedules.

The Open Risk Manual provides a modern, open and free collection of documentation resources covering all aspects of risk management.

The Open Risk Github Repository providing a collection of open source risk tools and risk models in support of the overall mission of Open Risk.

1 Solstice: Analytic Framework

This paper is part of the *Open Risk Solstice Manual*. We specify the analytic framework underpinning the Solstice simulation platform. This analytic design is what enables the simulation engine to deliver its functional requirements. This includes the modeling abstractions adopted, the representations and mappings of economic networks, the type of modeling that is enabled and the range of insights (metrics and reports) produced.

The objective here is to specify the framework in sufficient detail to illuminate how concrete implementations of Solstice in computer code are structured. Notation, formalism and further implementation details that are specific to use-cases and models are introduced as required in other parts of the Solstice documentation.

2 Overview of Core Components

The analytic framework has the following core components (laid out in subsequent sections).

- A representation of economic entities as *property graphs* (nodes and edges with associated attributes) as discussed in [1], [2]. Entities (3) are, on the one hand, characterized by the nature of their quantitative components and attributes (4) and relations with other entities (5) and, on the other, by the range of possible update operations.
- A representation of network-wide factors and dynamics drivers as *macro variables* (6)
- Sources of *uncertainty* (10) (both macro (system-wide) and idiosyncratic (entity-specific) (10.2).
- A discrete temporal grid (9) where future states of the economic network and (any) macro variables are modeled
- Network evolution along the temporal grid due to both deterministic and stochastic elements that follows by a variety of risk distributions / models (7) using potentially a combination of model components (satellite models) (12).
- Bottom up composition of network state changes using elementary ”system updates” (13).
- Ability to condition on specific realizations of (in particular) macro factors, which emulates scenario analysis / stress testing
- Introducing special entities that interact with the network in systemic ways (e.g. financial intermediaries, sovereigns or regulatory entities) with specialized internal states (16).
- Collection of sampling statistics of network evolution (15 and distillation of useful risk metrics and reports (17).

3 Network Entities

Entities are the actors (subjects) that populate the Solstice environment. They may represent individuals or legal entities acting in a legal capacity. Nodes will typically (but not necessarily) be economic entities but also various types of physical assets.

Computationally speaking entities are *objects* which derive their character from their associated component list (4) (their set of data attributes). This pattern is termed *composition over inheritance*. In accounting terms some entities might actually be reporting entities and thus some of the data characterizing them may originate in formal financial and non-financial reports.

While some entities can be assumed one-of-a-kind and play a special role in the network dynamics (Sec. 16), most are representative from a class of similar instances (though with different quantitative profiles). A separate functionality is through the use of *Pooled Entities* (homogeneous groups of entities). These are not modeled individually but rather through group properties (expressed e.g. as pool fractions).

4 Entity Components

Whereas an entity is described by an *X is a Y* type sentence, an entity component is anything that can be described by a *X has a Z* sentence. Components encapsulate the data attributes that entities have. These can be static characteristics or dynamic (evolving) state variables. Programmatically components are the object attributes or member variables of the entity objects (or node properties of network graph language). In financial accounts components would capture stock variables (permanent accounts) such as items listed in the balance sheet. In physical terms components might record physical states (position, velocity, temperature etc). Other examples of attributes are Business Model Information, Financials, Credit Risk Parameters etc.

This brings us to the *static description* of an economic network composition:

- V is a set (collection) of network nodes (entities). To accommodate entity diversity (entities with different attributes, capabilities and roles in the network) it is decomposed as $V = V^1 \cup V^2 \cdots \cup V^p$, where p is the total number of different *entity types*. For the network to exist there must be at least one entity type present (e.g. a set of corporate entities that form the counterparties to a bank loan portfolio).

- Different entity types will in general have varying number of components $n(p)$. Denote $C^p = \{c^1, c^2, \ldots, c^{n(p)}\}$ the set of n *entity components* of the p-th entity type. Such quantitative attributes are captured in one-dimensional entity component (vector / array) containers where the array index ranges over all entities belonging to this entity type. Different entity types may share a component class. For example both large corporations and small-and-medium size businesses may have components capturing accounting state.

- Collecting entity properties for all nodes of type p creates, in-principle, an entity component *data frame* $\mathbf{CD}^p$ that is, an $n \times N$ matrix of values ($n(p)$ columns, $N(p)$ rows) where $N(p)$ is the number of entities of a given type p.

Explicitly, for each entity type p, we have a set of tables such as:

i	c^1	...	$c^n(p)$
1	0.2	...	100
...	...	...	...
N	0.1	...	80

where i is the identity of each entity node. The table columns capture different attributes and the rows distinct instances of each entity type.

Open Risk
Solstice Manual:: Analytic Framework

5 Entity Relations

Next we explore the representation of *entity relations*. Network graph edges is the natural tool to express linkages, e.g., *economic relationships* between entities. Such relationships data can capture any concrete economic fact (transactions, contracts, shareholding interests etc.) that can be quantified into a number of attributes that can be assigned to edges. Economically relations show implicitly or explicitly in many places, such as in economic Input/Output (transaction) tables, shareholding networks, whom-with-whom statistics, supply chains, credit networks etc.

Relations can be both static and dynamic (with varying attributes). Relations express an X *is linked to* Y sentence. Whereas node attributes are stored in one-dimensional containers, relations are intrinsically *two-dimensional*. Formally:

- $E \subseteq V \times V$ is a set of edges connecting different nodes. The set of edges E is decomposed as $E = E^1 \cup E^2 \cdots \cup E^q$, where q is the total number of different *edge types*. Edge types may be node specific (that is applicable only to a combination of node types) or universal (able to connect all nodes).
- General relation data are represented as *edge attributes*. The complete network relationship structure is captured as $\bar{R}^q = \{r^1, r^2, \ldots, r^{m(q)}\}$, i.e. the set of $m(q)$ edge attributes of edges of type q. Different edge types q will in general have different number of attributes $m(q)$.
- The relations data frames $\mathbf{RD}^q$ are $m(q) \times M(q)$ matrices of values ($m(q)$ columns, $M(q)$ rows) where $M(q)$ is equal to the total number of edges (of a given type) that appear in a network.

The entity relationship data would look something like:

i	j	r^1	...	$r^m(q)$
1	2	0	...	1
...	...	...	...	...
N	N-1	0	...	0

A special case of relationships is provided by *adjacency matrices* A^q. Adjacency matrices are binary matrices consisting entirely of $(0, 1)$ values. The values A^q_{ij} where i and j are specific node entities indicates that a connection of type q exists between nodes i and j (but do not provide other qualitative or quantitative information). In standard mathematical graphs the relevant object is a single adjacency matrix. In most realistic economic networks one would have to introduce multiple edge types to track relevant connections. This can be thought of also as a set of matrices, or an *adjacency tensor* (A^m_{ij}) where m ranges over the different connection types. The adjacency tensor captures which *type of edge* exists between which pairs of nodes. Further complexity is introduced if permissible edge types are contingent on node types. In any case economic networks will in general have dynamic graph structure.

Putting together entity node components and their relations, diverse such entity and relationship collections are characterized as distinct *network shapes*. Each Solstice simulation handles one such network shape that is specified and initialized at the start of the calculation.

Open Risk
Solstice Manual:: Analytic Framework

6 Environmental (Macro) Dynamics

A Solstice story is a sequence of valid sentences that unfolds given the initial scene. Stochastic models are defined in terms of models / systems with computational and distributional (in the probabilistic sense) assumptions and embedded parameters which act on the entity components and relations elements we just defined. These stochastic elements produce alternative stories after the initial description of the network state.

The main object of interest in Solstice are the fate (e.g.welfare) of individual network entities. Yet it is seldom (but by no means impossible) that the entire dynamics of a network can specified without the need to introduce *external factors*. In general the network modeled is an *open system* in interaction with an environment that includes (in principle) the complement of whatever is *not* in the network.

In the Solstice framework macro information and modeling denotes any aggregate conditions that can affect the individual development of entities in the network. For example in an economic network this might capture key reference variables such as those are controlled by special entities in the network (e.g. interest rates), or observable signals that are generated via the aggregation of individual transactions (e.g. market prices of commodities) or even statistical aggregates about the network itself (or a super-set) compiled by statistical agencies (such as GDP).

At a given time t a given scenario realization of macro uncertainties is represented as the vector $\mathbf{Z}^t$. This is a dataset that captures all information that is not within the economic network state at that time point but is necessary for figuring its future states.

Effective modeling requires implementing correctly the interaction of the network with its environment, including any consistency constraints (e.g., conservation laws) or relevant feedback effects. For example, *decoupling* the dynamics of the environment from the network imposes some constraints on what size networks can be simulated consistently. Namely the implicit condition is that the size of the network compared to the entire system is such that it does not generate a feedback.

7 Simulation Models

The core deliverable of Solstice is a *simulation* of a specific economic network (a given configuration of entity types and relations) given a network evolution model.

The term Simulation denotes (in general language use) an organizational (management) process or tool that aims to explore and analyze a real-world situation or phenomenon (system, entity) to help inform decision making. This is achieved through the imitation (emulation) of the governing rules that are assumed to apply. The exercise aims to provide insights into the response of a system to varying future scenarios without having to actually wait for those to unfold (together with possible consequences).

Simulation models, in contrast with the *data models* we visited already and described as nouns, are *verbs*. They indicate actions by entities or other actors or the environment that lead to network state changes. They express dynamics and variation and they imply that some of the network attributes are *flow variables*. Programmatically they are implemented as functions or object methods acting on data (with possible parameters). In physical systems models are represented by the *equations of motion* (e.g. as derived in a Hamiltonian or Lagrangian formulation) while mathematically they are represented by ordinary / discrete differential equations (including stochastic components). In financial / economic systems models aim to capture the less well defined processes that modify, e.g. an organization's internal state and operations and may be represented e.g. by econometric equations.

Why do we need simulations? One reason: complexity. The systems we want to have may have both *observable complexity* expressed in the number of range of different attributes already present in the initial description of a network and *emerging complexity* from the large number of possible combinations of future events.

In many actual use cases simulation denotes a specific analytical concept (Monte Carlo simulation) that requires the use of well specified mathematical models from probability theory. These provide precise representations of the key characteristics or behaviors of the system or process being studied and are amenable to a variety of statistical and other quantitative analyses.

A Solstice simulation S is a bundle of data and methods characterized by:

- A configuration object that outlines the desired (high-level) calculation workflow
- A set of input resources (e.g. files or URL's) that specify and provide both economic network details
- The required model choices and model data that are postulated to govern the network dynamics (both deterministic and stochastic)
- The execution (running) of the simulation on a digital computer
- A set of (desired) outputs

8 Initial Setup and Subsequent Network Snapshots

An economic network initialization is setting up the scene and establishes its current state. This is composed by the data sets $\mathbf{V}^0 = \{\mathbf{CD}^{p,0,0}, \mathbf{RD}^{q,0,0}\}$ where $t = 0$ is by convention the initial time. Effectively, the initialization process picks up a snapshot from a network's past evolution without necessarily encoding knowledge of how it came about.

Dynamic economic networks can be modeled most simply as a sequence of static graphs. Each snapshot of the sequence represents the network entity states and their relations at that given time. This is composed of the sets $\mathbf{V}^{t,s} = \{\mathbf{CD}^{p,t,s}, \mathbf{RD}^{q,t,s}\}$. For concreteness, these data sets are essentially collections of

1. one-dimensional variable vectors $\mathbf{C}^{p,t,s} = \{C_i^{p,t,s}\}$ and
2. two-dimensional matrices respectively $\mathbf{R}^{q,t,s} = \{R_{ij}^{q,t,s}\}$

The detailed simulation typically takes place in a broader macro environment so the evolution of the network might be split into two steps (macro then micro), unless there are feedback loops, in which case further iterations might be necessary. In all cases, the collection of objects $\mathbf{V}^{t,s} = \{\mathbf{C}^{p,t,s}, \mathbf{R}^{q,t,s}\}, \mathbf{Z}^t$ captures the essential network configuration at future times t.

9 Temporal Grid

Future states of network entities and macro factors are considered at given **timepoints** $t \in [T_0, T_M]$. The period spans the time interval from the current date T_0 till the calculation horizon (longest duration) T_M of all calculations.

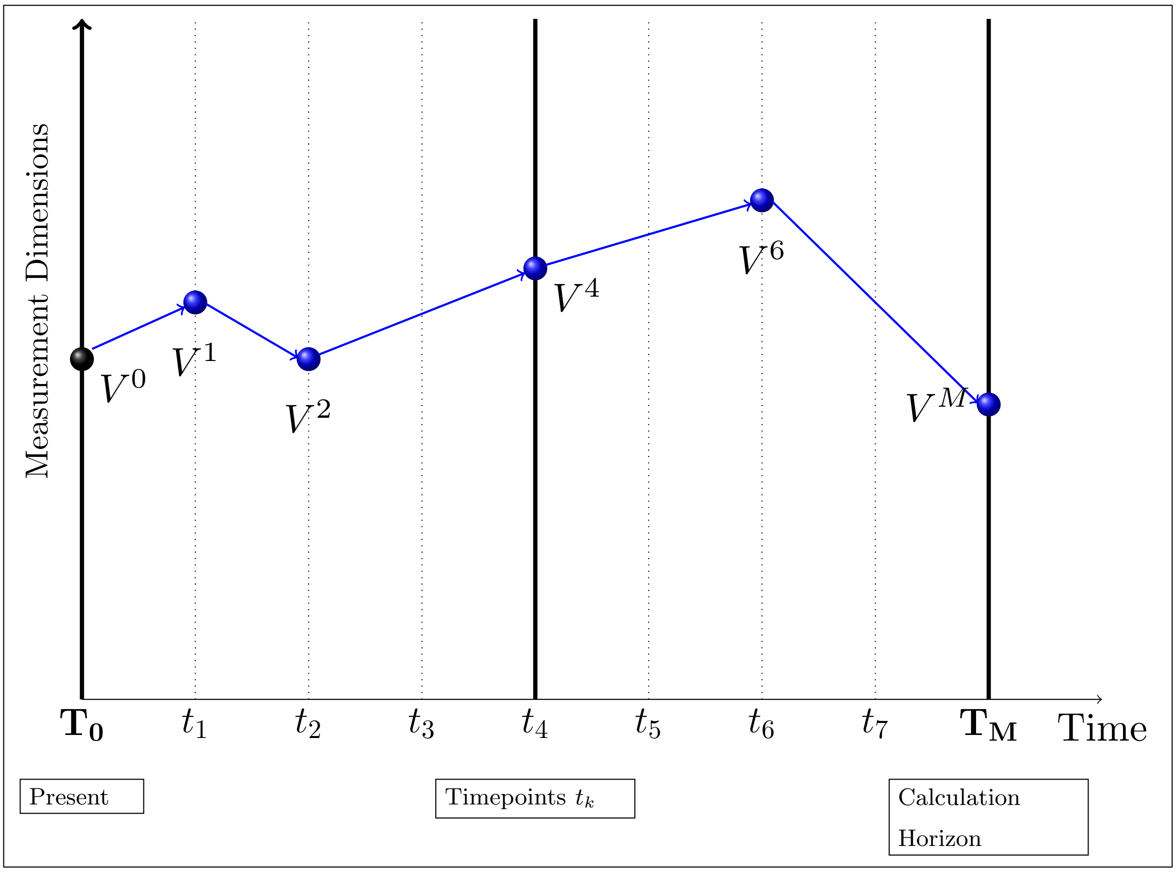

Figure 1: The diagram illustrates the temporal discretization underlying the calculation framework of the Solstice. All quantities V_t modeled within the framework refer either to the timepoints t or the periods between two timepoints $[t-1, t]$. The present time (t_0) is special in that all quantities are assumed known.

NB: All variables are either *state* or *flux* (flow) variables. State variables are associated with a single timepoint t whereas flow variables are associated with a period $[t_-1, t]$.

10 Sources of Uncertainty

It is already the case that networks with entirely deterministic dynamics may exhibit complex behavior (and hence require computational tools to explore). But the raison-d-etre of Solstice it to help explore the impact of *sources of uncertainty* (in particular *risks*) in various economic network structures. Uncertainty is thus approached as an exogenous influence on the network and Solstice is using a standard *Monte-Carlo framework* based on pseudo-random numbers generated digitally.

At any timepoint a set of simulated **random variables** captures the nature of uncertainties that are recognized with the modeling framework. Such random variable realizations may be recurring (stochastic processes) or one-of events. They may be associated with individual entities or apply to the network at large (macro factors). They might be either continuous or discrete random variables.

These conjectured realizations of key stochastic variables "shape" the stochastic environment (condition on certain outcomes). A single distinct scenario $s \in [1, S]$ is a collection of realizations for all underlying stochastic factors. Scenarios are intrinsically multi-period and must match the desired calculation horizon.

An economic network update model has the general form:

$$\mathbf{V}^{t,s} = M_V(\mathbf{V}^{t-\tau,s}, \mathbf{Z}^{t-\tau,s}, \epsilon^{t-\tau,s}) \; \tau \in [0, \infty) \tag{1}$$

$$\mathbf{Z}^{t} = M_Z(\mathbf{Z}^{t-\tau,s}, \xi^{t}) \tag{2}$$

where $\mathbf{Z}$ are macro scenarios (optionally subject to uncertainty) and potentially other background information that is not embedded in the entity network . In principle any number of prior periods might (as captured by the range of τ may be required for calculating the new network state at time t. In the above we introduced also the symbol $\epsilon^{t-\tau,s}$. This represents *idiosyncratic uncertainty*, whereas ξ^t captures system-wide uncertainty.

10.1 Example: Macroeconomic VAR Processes

A salient feature of macro environments that is important to include in the framework is the empirical fact that macro factors do not evolve as pure random walks but exhibit *auto-regressive* behavior. A fairly general expression for the process followed by observable macro factor vector $\mathbf{Z}^t$ is the standard vector-autoregressive VAR(p) process [3]

$$\mathbf{Z}^{t} = c + \sum_{\tau=1}^{p} \phi_\tau \mathbf{Z}^{t-\tau} + \xi^{t} \tag{3}$$

where $\xi_t \sim N(0, \Omega)$ are macro uncertainty sources with embedded model dependent distributional assumptions and c, ϕ_τ are model parameters.

10.2 Idiosyncratic Uncertainty

Idiosyncratic uncertainty ϵ is a modeling tool that enables exploring the impact of random events at the individual entity level. This is particularly relevant when entities have inhomogeneous attributes (e.g., widely varying sizes)

11 Model Families

A model family is the collection of *systems* and parameters that express the network update. Solstice is agnostic about which model to apply. A list of possible models includes:

- Markov Chain based Simulation
- Simulations using Copula Methods
- Hazard Rate Models
- Non-Linear (Threshold) Models
- Simulated Contagion Models
- Cellular Automata Models

Open Risk
Solstice Manual:: Analytic Framework

Each model family has as set calculation methods and possible flags, parameters etc. that select the specific type of model to be run. Given the impact of model assumptions on the outcomes, models should be as simple and transparent as possible. In particular there should be ownership by users of the parameters and distributional assumptions involved.

12 Model Components

Model Components are data pieces that can be pulled together to help implement a particular model. Typical model component data might be the parameters of various satellite models e.g., autoregressive process parameters, correlation matrices etc. Such model components might be shared between several Solstice models families. They might also be reused in several systems per model.

Model components need (in general) to be estimated. Such estimation is done outside Solstice and will typically involve fitting suitable functions to historical data (e.g. past events in representative pools of entities). They represent thus in general global behavior characteristics of entity types.

12.1 Counterexample: Analytic Models

Analytic models are a very special class of probabilistic models that are amenable to analytic calculation (expressed in terms of relatively contained mathematical expressions / functions). Hence they bypass the requirement to sample scenarios and compute statistical outcomes via simulation. Instead the distributional properties of future states of the network are inferred from these analytic estimates.

Analytic models are general only feasible when the stochastic elements associated with the network can be captured in a small number of random variables with tractable joint distributions.

13 Computation Systems

Whereas a Model encapsulates *all* the action that needs to take place to produce a network update from time t to time $t+1$, a *System* is the smallest possible self-contained update action. Computation Systems are thus the fundamental building blocks of Models.

Systems are essentially *functions* acting on a subset of network data as of time t (and potentially prior times) and producing intermediate data or final updated states for time $t+1$. For example acting on entity components (in possible combination with macro factors) and producing updated values for these components as indicated by the equation:

$$\mathbf{C}^{p,t+1,s} = S(\mathbf{C}^{p,t,s}, \mathbf{Z}_t) \tag{4}$$

A model can be seen as a function composition of one or more systems applied in sequence. Symbolically:

$$M = S_k \circ \ldots \circ S_1 \tag{5}$$

14 Model Calculation Outline

A Solstice model family is characterized by a calculation flow that looks as the following pseudocode:

Open Risk
Solstice Manual:: Analytic Framework

```
MODEL:
ONE OR MORE SCENARIOS:
    ONE OR MORE TIMESTEPS:
        ONE OR MORE SYSTEM BLOCKS:
            SYSTEM(NETWORK DATA, MACRO DATA)
```

The above basic structure can materialize in substantially different configurations: Many scenarios for one timestep, many timesteps for one scenario and all intermediate options.

15 Insights

As discussed in Sec 8, the collection of objects $\mathbf{V}^{t,s} = \{\mathbf{C}^{p,t,s}, \mathbf{R}^{q,t,s}, \mathbf{Z}^t\}$ captures the essential network state at all simulated times and all scenarios. The objects in $\mathbf{V}^{t,s}$ are in-general too complex to be of immediate use or even inspection. One source of complexity is their high overall dimensionality: 1) the potentially large number of different attributes characterizing entities and their relations, 2) the typically large number of entities (and hence relations) in real networks, 3) the number of timesteps - which may be significant if one desires e.g. accurate temporal resolution and, finaly, 4) the number of alternative scenarios examined which (for statistical based analyses) must be based on a sufficient number of samples.

Insights are created by applying statistical analysis on the network data. The raw data produced during the simulation must be processed to extract the most interesting or useful pieces of information. This is an open-ended task that depends on the situation being simulated and the objectives. The general form of this exercise can be stated as producing a set of (quantitative) insights $\mathbf{I}$:

$$\mathbf{I} = G(\mathbf{C}^{p,t,s}, \mathbf{R}^{q,t,s}, \mathbf{Z}^t) \tag{6}$$

The ultimate insights may be various risk indicators and metrics, flexible filters or other aggregations that have an explanatory (e.g. visualization oriented) or validation character.

The different types of distributions extracted from the simulation can broadly be grouped into discrete and continuous distributions. For example the distribution of entities in different discrete states (e.g. ratings) will be a discrete distribution. On the other hand the market value of entity assets will be a continuous distribution.

Let us focus for simplicity on a single entity component type and the associated generated data set. Let $C_i^{t,s}$ denote the entire sampled data set, which is three-dimensional, ranging over all entities i possessing this attribute, all iterated timesteps t and all applicable scenarios s.

The types of useful statistics that can be produced depend on the relative role of these dimensions. The general strategy is to reduce the dimensionality by conditioning. Let examine some important options in turn.

15.1 Population Statistics

Population statistics are *distributions*, denoted $F_c^{t,s}$, per component c answering the question of *how are entity properties distributed at a given time t and conditioning on a scenario s*. Depending on the nature of the component / attribute, the distribution might be continuous or discrete, have finite or infinite range etc. Moments or other risk metrics can reduce this distribution to easier to communicate scalar values (e.g. average loan size over time across a baseline scenario, a growth scenario and degrowth scenario).

Population statistics make sense also when there is a single scenario and/or a single calculated time step. They do require a relatively large number of entities before being reliable from a mathematical perspective. Augmenting the number of properties examined leads to higher dimensional (joint) distributions $F^{t,s}_{c1,c2}$. This enables, for example, examining the correlation of various population properties across the network.

15.2 Unconditional Scenario Statistics

Finally, distributions of properties over scenarios, denoted $F^{i,t}_c$, answer the question of how network entity properties change after one (or more) evolution steps over a (potentially large) number of different scenarios. Here too, while such scenario distributions may be interesting per individual entity, they will typically be more useful as *aggregate* network properties (e.g. averages). For a distributional approach to make sense, the number of scenarios must be relatively large. In any case scenario realizations must have an explicitly or implicit assigned probability.

A *Risk Horizon* T_H is any future timepoint at which the overall external and internal state of the network is assessed to produce scenario statistics. There is no optimal selection of a risk horizon. Its value is a user option in configuring the Solstice framework and may vary in different use cases. In concrete usage of course that flexibility must be reduced by asserting which of the choice is most representative or useful. Guidance for selecting the risk horizon is provided by the timescale over which the network exhibits significant change.

15.3 Temporal Statistics

Temporal statistics explore another dimension of the sampling data set, in a context where there is an significant number of evolution timesteps. Denoted $F^{i,s}_c$, this distribution answers the question of how properties vary over time for a *given* entity i and *conditioning* on a scenario s.

The focus on a specific entity may be interesting in itself but frequently a further reduction is useful, such as examining population or scenario averages.

16 Managers and Their Objectives

Managers are a special type of entity in Solstice that represent agents such as portfolio/risk managers, regulators, public authorities etc. A Solstice user may (but need not) be acting on behalf of such a manager in order to support their mandate.

Managers in Solstice are general entities that interact with (and may be affected by) the developments in the economic network. For example a portfolio manager may have explicit dependency on the state of their portfolio (profitability, solvency, emissions intensity etc) which is a subset of the economic network.

Manager objects are the natural tool to implement and explore how e.g., portfolio management strategies perform under various scenarios. These interactions are not standardized but must be implement in each instance to reflect the specific requirements.

17 Risk Metrics, Tabular and Visual Reports

Solstice is not a reporting platform but it produces quantitative datasets in formats that allow easy integration in modern reporting workflows. The details of reporting options are documented elsewhere.

Open Risk
Solstice Manual:: Analytic Framework

Here we highlight what we mean by "reports" in the context of the Solstice analytic framework.

In Sec.15 we saw that the main mechanism for deriving insights from Solstice simulations is via the various distributions being produced. Data gathering objects such as $F_c^{t,s}$ (scenario specific population statistics) or $F_c^{i,t}$ (unconditional statistics) provide the raw material for generating a vast variety of possible reports. It is not in-general desirable to report the entirety of these objects. Despite the dimensionality reduction (e.g. conditioning on scenarios, time slices etc) they may still include significant number of variables. The strategy thus involves further simplifications (the precise nature of which is context dependent). Examples:

- In the first instance informative reports may based on constructing *summary measures* that reduce the distributions F_c to a few scalar values. The classic example are risk measures such as Value-at-risk computed from the shape of (one-dimensional) distribution.
- Tabular reports are in their simplest form simple extracts (filtered subsets from F_c): E.g., the top-ten highest losses in an adverse scenario.
- Visualization reports on the other hand pursue a different strategy: instead of filtering / summarizing large volumes of quantitative data, they allow ()high-level) overview by constructing visual (graphical) representations. Classic tools in this context are the different types of probability plots.

18 Forward Expectations

Forward expectations are an example of more sophisticated analyses of future events which play a key role in some economic network analyses. Mathematically and conceptually forward expectations are formed at a given time t by averaging the projected outcomes of random variables under different scenarios (with different probabilities) given an information set available at that forward time t.

When the risk horizon T_H is less than the final horizon (T_M) there might be residual uncertainties that have not been resolved by that time. This gap can can be addressed by a number of distinct approaches

- Perfect Foresight Models: A single scenario for residual uncertainty realization which may be the most likely scenario or a stressed scenario
- Expectation Models: Multiple probability weighted scenarios for residual uncertainty realization

In principle forward expectations can be computed and used alongside regular state variables at any timepoint. Yet this becomes a progressively more expensive calculation the more timepoints it is applied to. In practice Solstice is adopting a compromise solution where (when the use case requires its) forward expectations of events up to the calculation horizon are estimated at the intermediate risk horizon. This is essentially a two-period framework. We use the terminology *Root and Branch* scenarios.

18.1 Computation based on Root and Branch Scenarios

To enable the computation and use of forward expectations, the set of total simulation scenarios S is organized in groups of *root or primary* and *branch or additional* scenarios ($S = R \times B$).

A root scenario is a multiperiod realization of uncertainty factors that starts at the current time $t = 0$ and extends till the calculation horizon T. A branch scenario is identical with its root scenario until

Open Risk
Solstice Manual:: Analytic Framework

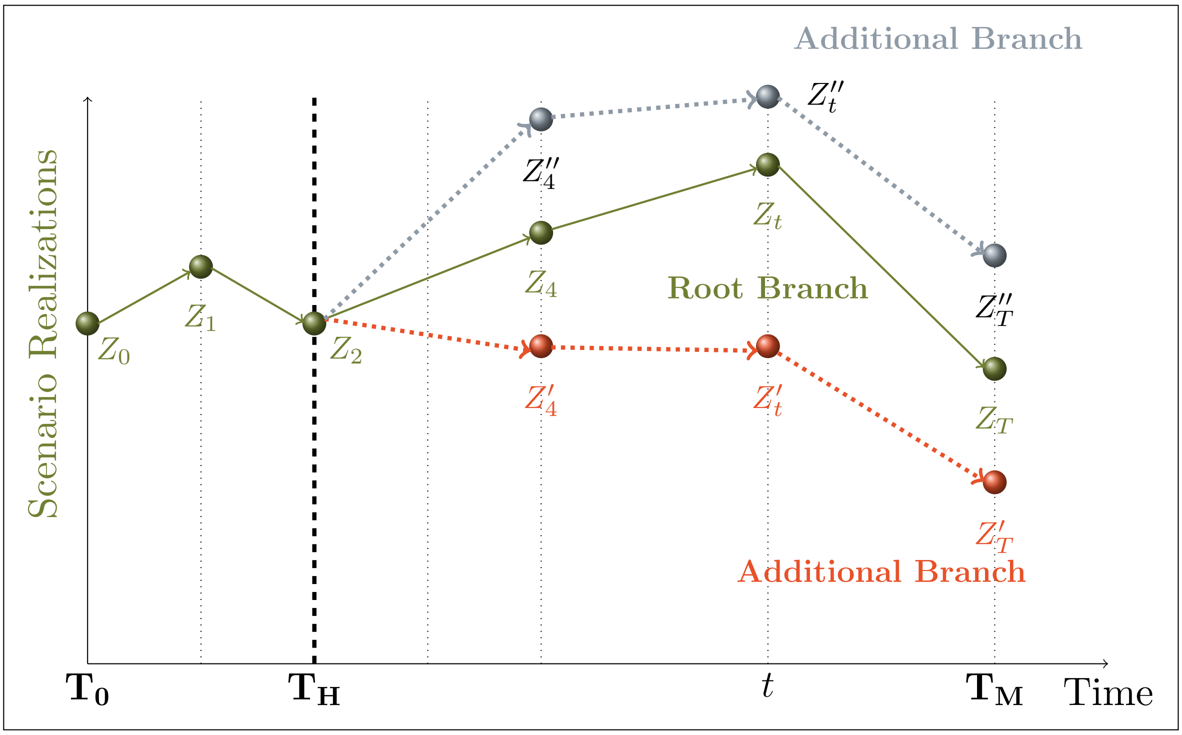

Figure 2: The diagram illustrates the organization of scenarios into a root scenario extending till the calculation horizon T_M and additional branch scenarios that explore alternative realizations of future scenarios post the risk horizon T_H

the risk horizon but may branch off subsequently. Expectations at the risk horizon T_H for each root scenario are calculated by averaging the corresponding branch scenarios (Fig.2). Within Solstice such expectations are calculated consistently on the basis of the postulated underlying processes and do not introduce additional sources of uncertainty.

18.2 Example: IFRS 9 / CECL based portfolio management

An important example of the role of forward expectations is offered by the integration in network analysis of credit risk *accounting systems* such as IFRS 9 or CECL. These standards require that comprehensive credit risk information must incorporate not only past due information (hence in our terminology the network state at some future point t but also all relevant credit information available at that time, including *forward-looking* macroeconomic information. In this context various relevant expectations can be computed: Credit Default Expectations at the Risk Horizon, Credit Recovery Expectations at the Risk Horizon and Composite metrics such as **Expected Credit Loss** and Loss Allowances.

Open Risk
Solstice Manual:: Analytic Framework

References

[1] P. Papadopoulos. WP8: Connecting the Dots: Economic Networks as Property Graphs. *Open Risk White Papers*, 2019. Online Link.

[2] P. Papadopoulos. WP10: Connecting the Dots: Concentration, diversity, inequality and sparsity in economic networks. *Open Risk White Papers*, 2021. Online Link.

[3] J.D. Hamilton. *Time Series Analysis*. Princeton University Press, 1994.