DAFNE

A **D**ecision-**A**nalytic **F**ramework to explore the
water-energy-food **NE**xus in complex and transboundary
water resources systems of fast growing developing countries

A DISTRIBUTED HYDROLOGICAL MODEL TO SIMULATE HYDROLOGICAL RESPONSE, TRANSPORT PROCESSES AND SEDIMENT DYNAMICS

Deliverable D3.1

August 2018

EU H2020 Project Grant No. 690268

Programme Call: Water-5-2014/2015
Project Number:.....690260
Project Title:.....DAFNE

Work-Package:.....WP3
Deliverable #: D3.1
Deliverable Type: .....Document
Contractual Date of Delivery: 31 August 2018
Actual Date of Delivery: .....31 August 2018
Title of Document: .....A Distributed Hydrological Model to Simulate Hydrological Response, Transport Processes and Sediment Dynamics
Author(s): .....Enrico Weber, Giulia Battista, Martina Botter, Scott Sinclair, Paolo Burlando

Availability: .....Public report

Document revisions		
Author	*Revision content*	*Date*
Scott Sinclair	Complete compilation of first draft for review by quality managers	2018-08-16
Jos Van Orshoven	Review	2018-08-23
Paolo Burlando	Review	2018-08-25
Scott Sinclair	Changes based on review, and finalize	2018-08-28

Table of Contents

List of Tables

List of Figures

Abbreviations

AAIGrid: ESRI Arc/Info ASCII Grid (files)
API: Application Programming Interface
CF: Climate and Forecast (metadata compliant file format)
CSV: Comma Separated Value (files)
DoA: Description of Action (Annex I of the Grant Agreement)
ETHZ: Eidgenössische Technische Hochschule Zürich
MCT: Muskingum-Cunge-Todini (flow routing algorithm)
ODE: Ordinary Differential Equation
OTB: Omo-Turkana Basins
PET: Potential EvapoTranspiration
SAS: StorAge Selection (function)
SSC: Suspended Sediment Concentration
SST: Suspended Sediment Transport
TOPKAPI: TOPographic Kinematic APproximation and Integration (model)
TOPKAPI-ETH: The extended implementation of TOPKAPI developed at ETHZ
TT: Transit Time
TTD: Transit Time Distribution
UNZA: University of Zambia
USLE: Universal Soil Loss Equation
WATET: Water Age and Tracer Efficient Tracking (model)
WEF: Water, Energy and Food (nexus)
WP: Work Package (e.g. WP3)
ZRA: Zambezi River Authority
ZRB: Zambezi River Basin

1. INTRODUCTION

In this report we present the work underway to re-engineer an existing physically-explicit hydrological model (TOPKAPI-ETH, *Fatichi et al.*, 2015) in order to meet the modelling demands imposed by the DAFNE Water Energy and Food (WEF) nexus modelling framework. The extended and redesigned hydrological model will form the flexible core component of the integrated WEF nexus model of deliverable D3.5 due for delivery in month 36. In parallel, to support the programming developments, we have been carrying out research and development activities to include new processes in the model for the spatially distributed simulation of solute and sediment transport. These processes are strongly controlled by agricultural expansion and operational practices, as well as the construction and operation of reservoirs for hydropower and water supply, thus being a necessary model component of the integrated WEF nexus model.

To give context to our approach in this report, the deliverable description from the DoA is highlighted in the box below:

> *"D3.1 – A distributed hydrological model to simulate hydrological response, transport processes and sediment dynamics (HWRM-ETHZ, M24).*
>
> *Description of the modified distributed hydrological model accounting for transport processes and sediment dynamics in spatial and temporal explicit fashion."*

As required by the DoA, we focus on the description of the model modifications (and justification for our choices). This report provides a description of the modified model along with the current and planned components under implementation. This introductory section provides some context for the work. Then, in the remainder of the report we outline the hydrological transport component that will be used to simulate the water quality dynamics at the river basin scale with respect to solutes, either mobilised by water flows or by sediment transport. These are core active research and development activities under WP3 (section 2). The implementation of these transport related water quality components into the new TOPKAPI-ETH version is one of the goals of the activities around this deliverable (D3.1) and the upcoming D3.5. We also describe the software development work underway to redesign the TOPKAPI-ETH hydrological model, making it more computationally efficient, flexible and portable (section 3). In section 4, we detail the progress to date, and our next steps as we build the hydrological model and work to integrate and couple the other modelling components of DAFNE (e.g. Lake process modelling MS22; Agricultural modelling MS27). In this respect, while the redesign of the code is ongoing, the model conceptualisation has been completed. Finally, in section 5 we highlight the main conclusions from this component of work and how this fits into the DAFNE project as a whole.

1.1 CONTEXT

According to the 2018 United Nations World Water Development Report, the world's population is expected to increase from 7.7 billion in 2017 to between 9.4 and 10.2 billion by 2050 and more than half of this anticipated growth is expected to occur in Africa (+1.3 billion). Consequently, global demand for food and energy production, both of which are water-intensive, is expected to increase by roughly 60% and 80% respectively by 2025 [*WWAP*, 2018]. In this context multi-disciplinary planning tools capable of both qualitatively and quantitatively evaluating the trade-offs between development pathways have great value. Such comprehensive tools are currently lacking, mostly due to the challenges of integrating trans-disciplinary knowledge into a coherent framework [*Albrecht*, 2018].

In the Zambezi River Basin (ZRB) energy production and agriculture are two of the main anthropic activities which have a significant impact on the water resources availability and on the biogeochemical response of rivers [*Gall et al.*, 2012; *Benettin et al.*, 2017]. Dams for hydropower pur-

poses result in a discontinuity in the solute transport along the river network, which alters the naturally occurring solute and sediment transport, deposition and transformation processes. Indeed, substances carried by rivers and surface runoff enter the reservoirs, where the reduction in transport capacity and long residence times result in the substances coming out of suspension or solution. The resulting deposition and other biogeochemical processes impact the water quality of downstream river reaches. The existing hydropower infrastructure, comprising the four largest hydropower reservoirs in the ZRB (Itezhi-Tezhi, Kafue Gorge, Kariba and Cahora Bassa) exploit less than one-third of the total hydropower potential of the basin [*Lautze et al.*, 2017]. Given this large potential for further hydropower development, water resources management authorities should consider both water availability and water quality. According to *Lautze et al.* (2017) irrigated agriculture will increase from here to 2025 increasing the share of the total water use from 1.43% to 4.49%. Nevertheless, food security continues to be an issue, especially in Africa, and, under the pressure of climate change, it will be one of the main challenges for the next decades. Fertilisers and plant protection product applications play a crucial role in the increase of agricultural yields, but massive fertiliser inputs cause ecological problems, and thus awareness about agricultural nutrient management must be an important consideration of scenarios describing agricultural expansion in developing countries like the ones in the ZRB.

The second DAFNE study area, the Omo-Turkana River Basins (OTB) is also undergoing rapid changes and developments in the present times. Although these developments are relatively new, in contrast to the developments in the ZRB.

The Omo River has a strategic importance in the Ethiopian water resource balance and hydropower potential, being the second biggest river by discharge volume in Ethiopia after the Blue Nile. In 2004, the construction of a cascade of dams began with the commissioning of the Gibe I reservoir and hydropower plant. In 2009 the Gibe I scheme was enhanced by adding a tunnel to the Gibe II hydropower plant; then, in 2016, Gibe III, the biggest of the cascade of dams started operating. The cascade will be completed with the Koysha dam that is at the moment under construction [*Avery*, 2012] and replaces the previous plan to construct two more dams, Gibe IV and V.

Large-scale irrigation projects are the other recent development in the country. In early 2011, the Omo-Kuraz large-scale sugar plantation began development along the Lower Omo, just upstream of the Omorate, which is closest cross-section to the outlet into Lake Turkana where flow data are known to exist [*Avery*, 2012].

The combined effect of the hydropower dams and irrigated agriculture on the recession agriculture practices on the banks of the lower Omo, as well as fisheries and livestock activities surrounding the endorheic lake Turkana in Kenya is the subject of significant ongoing tension in the region.

Since future projections suggest an increase of both agricultural area and hydropower production in the Omo and Zambezi catchments, it is reasonable to expect that the solute transport dynamics will be affected making the understanding the impacts of these developments on the water quality an issue that deserves high priority.

Below we describe the developments of the modelling tools that will be adopted to investigate the hydrological processes and their interaction with the transport of sediments and solutes at the catchment scale. After a brief overview of the state of the art of the scientific background in terms of hydrological, sediment and solute transport, we will explain the details of the adopted modelling tools.

2. WATER QUALITY COMPONENTS

When using the term water quality in this report, we refer specifically to dissolved solutes, or sediments transported either in suspension or as bed load. In this section we first give a brief introduc-

tion to the current state-of-the-art before outlining the research direction and planned implementation strategies for solute and sediment transport respectively. The motivation for including these processes is outlined in section 1.1.

2.1 STATE OF THE ART

In the last decades, the study of river basin hydrology has generally evolved towards a comprehensive theory describing water and energy exchanges between land surface and atmosphere at several scales [*Rigon et al.*, 2005]. Various distributed watershed models have been developed which implement these concepts and to achieve integrated management of water resources.

Some of these models have developed towards the inclusion of the fluxes of sediments and solutes within the catchment. The simulation of these fluxes, as a function of the solution of the hydrological fluxes within a spatially distributed model, provides a means of predicting the natural and anthropogenic influences on water resources at multiple locations within a catchment [*Shen and Phanikumar*, 2010].

A number of physically-explicit numerical models exist that describe these fluxes at the catchment scale. However, their typically high computational cost and the detailed input data requirements are usually a major constraint to the application of these tools to model larger catchments.

In the context of the DAFNE research project we aim at developing a physically-explicit distributed hydrological model including the description of solute and sediment fluxes that is computationally efficient and thus allows the application of a distributed modelling approach for large-scale catchments such as the Zambezi.

We start from the framework of the physically-explicit, spatially distributed hydrological model TOPKAPI-ETH [*Fatichi et al.*, 2015]. The TOPKAPI-ETH model is particularly efficient from the computational point of view because the non-linear reservoir equations, stemming from the kinematic approximation of the flow equations, are solved analytically, using approximations that are valid for a wide range of conditions [*Ciarapica and Todini*, 2002].

This approach makes the model particularly suitable to work on very large basins while explicitly modelling the connectivity of solute and sediment fluxes driven by the fluxes of water.

2.1.1 Solute transport modelling

Since the 1970s a branch of hydrology has moved its focus from global mass-balance catchment understanding to the study of the individual components of the terrestrial water cycle, which might have a controlling influence on the cycling of solutes, contaminants and nutrients [*Botter et al.*, 2010; *McDonnell*, 2017].

On the one hand, the concepts of water travel time distribution and water residence time have been widely explored, since then, in order to derive understanding about the storage, geochemistry, flow pathways, sources and sinks of water [*McGuire and McDonnell*, 2006; *Botter et al.*, 2010; *Godsey et al.*, 2010; *Benettin et al.*, 2013]. These aspects have often been addressed by focusing on the transit time (TT), which is the time required to rainwater to reach the stream and the distribution of which reflects the diverse flow paths that water from rainfall can take before arriving in the channel [*Kirchner et al.*, 2001]. Isotopic and environmental tracer data have proven to be valuable candidates for evaluating Transit Time Distributions (TTD, *Kirchner*, 2006; *Soulsby*, 2009; *McDonnel et al.*, 2010; *Beven*, 2012), thanks to their non-reactive nature. Indeed, as they move through the catchment, from the source to the outlet, they are not altered by chemical or biological processes. Therefore, their export distribution can be considered a valuable descriptor of the water pathways across the catchment. Different tracer-aided models have been developed in the last decades [*Soulsby et al.*, 2015]. Initially, TTD representations described the integral behaviour of tracer transport through the catchment (using a convolution integral approach), assuming an *a priori* time-invariant TTD [*McGuire et al.*, 2005; *Tetzlaff*, 2009]. Subsequently, time-variant TTD theory was introduced based on lumped conceptual models [*Botter et al.*, 2010; *Botter et al.*, 2011;

Rinaldo et al., 2011; *van der Veelde et al.*, 2012; *Benettin et al.*, 2013; *Harman*, 2015]. Most recently, the time-variant TTD concept was developed into the StorAge Selection (SAS) function theory. While time-variant TTDs represent the probability distribution of the transit time of water particles, the SAS function determines the criterion of sampling water from the catchment age-ranked storage, thus representing the relationship between the set of ages available in the storage and the age of the particles removed as outflows.

On the other hand, little progress has been made in modelling spatially explicit solute fluxes [*Birkel and Soulsby*, 2015; *Soulsby et al.*, 2015]. Although physically based models are highly dependent on parametrisation and are computationally costly, further research effort should be dedicated to the development of spatially distributed descriptions of solute transport processes, because these models have high potential to represent the dynamics of reactive substances at multiple locations within a catchment [*Hrachowitz et al.*, 2016]. Indeed, a spatially distributed description of the transport processes does not require any *a priori* assumption about the shape of the TTD, since the tracer can be tracked in each discretized element of the domain for each time step of the simulation.

The coupling of hydrological transport processes and non-conservative solute export at the catchment scale is particularly challenging, since predictions of in-stream concentrations are often confounded by unknown inputs and by low-frequency sampling. Non-conservative solutes undergo several different transformation processes along their pathway from the source to the outlet, which might be physico-chemical (e.g., temporal mobilisation), bio-physical processes (e.g., plant uptake, biological fixation) or reactions (e.g. sorption, mineralisation, volatilisation). In the recent years' research moving in this direction has focused on developing concentration-discharge (C-Q) relationships [*Godsey et al.*, 2009; *Basu et al.*, 2010; *Moatar et al.*, 2017; *Wymore et al.*, 2017] as clues to the hydrochemical processes that control runoff chemistry [*Godsey et al.*, 2009]. In a log(C)-log(Q) space, C-Q relations have been observed to be linear in many cases [*Godsey et al.*, 2009], so that the empirical relations can be well approximated by a power-law, $C = aQ^b$, where a and b are fitting parameters [*Godsey et al.*, 2009; *Basu et al.*, 2010; *Thompson et al.*, 2011; *Moquet et al.*, 2016; *Moatar et al.*, 2017; *Musolff et al.*, 2017]. A very common metric, relevant also for this study, is based on the value of the b exponent, the slope of the regression in the log(C)-log(Q) plot, because it is related to the concept of "chemostasis" [*Godsey et al.*, 2009] or "biogeochemical stationarity" [*Basu et al.*, 2010]. A catchment shows "chemostatic" behaviour when despite a sensible variation in discharge, solute concentrations show a negligible variability, i.e., $b \cong 0$. Conversely, positive slopes (i.e., increasing concentrations with increasing discharge) would support an enrichment behaviour where the solute amount grows with discharge and negative slopes (i.e., decreasing concentrations with increasing discharge) support a dilution behaviour with solute mass that does not increase proportionally to the growing discharge. A solute is typically defined as transport-limited if it is characterized by enrichment, while it is called source-limited in case it dilutes [*Duncan et al.*, 2017]. Besides C-Q relations, some physically-based models, instead, simulate hydrological processes and nutrient transport in the different compartments of the catchment. Some of the most known models are ANSWERS [*Beasley et al.*, 1980], SWAT [*Arnold et al.*, 1993], HSPF [*Bicknell et al.*, 2001] and AGNPS [*Young et al.*, 1989]. Although they can provide accurate results, they require a large number of parameters, which often cannot be directly measured in the field and they are also demanding in terms of computational time. To circumvent this limitation, other models describe hydrological and solute transport processes in a semi-distributed way and they usually use Hydrological Response Units (HRUs, *Sharpley et al.*, 2007; *Ding et al.*, 2010). Nevertheless, there are very few studies that have attempted to model both flow and water quality of non-conservative tracers in a consistent way at the catchment scale, although this is a promising direction for future work [*Beven*, 2012; *Hrachowitz et al.*, 2016). A coupled hydrological-water quality model for solute concentration prediction at the outlet is used in a few studies [*Botter et al.*, 2006; *van der Velde et al.*, 2010], but the issue of lacking input data is not exhaustively solved, since in both studies nitrate concentrations at the river outlet are estimated based on the hypothesis of downscaled input from an average annual value to a daily value, which is necessarily based on many assumptions.

2.1.2 Sediment transport modelling

A first concentrated effort at soil erosion prediction was the Universal Soil Loss Equation (USLE), an empirical formula relating mean annual erosion rates to soil properties, conservation practices, land cover, slope and climate through a number of empirical parameters [*Wischmeier and Smith*, 1978].

A more process-based approach was initiated by the work of *Foster and Meyer* (1972), who proposed to compute erosion and deposition based on the difference between transport capacity and sediment flow rate along hillslope flow paths (e.g. the Revised-USLE, *Renard et al.*, 1991).

Following the work of *Foster and Meyer* (1972), many physically based numerical models were proposed in the literature (see *Merrit et al.*, 2003 and *Aksoy and Kavvas*, 2005 for reviews). Those models use sediment transport and erosion formulas for hillslopes and river channels based on the concept of transport capacity (e.g., *Beasley et al.*, 1980; *Mitas and Mitasova*, 1998; *Molnar et al.*, 2006).

Many of the earliest numerical models focused on predicting soil erosion rates at the field scale, for land management applications and water quality predictions. Models like ANSWERS [*Beasley et al.*, 1980] and CREAMS [*Knisel*, 1980] for example, provide a detailed description of hillslope processes, however, they miss a component that routes sediments through the channel network to the outlet of the catchment.

Later models complete the components necessary for modelling both sediment production and routing: a rainfall-runoff module, a hillslope erosion module and an in-stream transport module. Some of those models still contain many empirical or conceptual approaches – MIKE-11 [*Hanley et al.*, 1998], HSPF [*Walton and Hunter*, 1996] – while others moved towards a purely physically based description of the problem.

WEPP [*Nearing et al.*, 1989], SHESED [*Wicks and Bathurst*, 1996] and EUROSEM [*Morgan et al.*, 1998] are three examples of physically based erosion and sediment transport models developed in Europe and in the U.S. These models contain a highly detailed description of the hillslope processes; however, this comes at a high computational cost and with onerous input data requirements. These two factors limit the application of the models to the field scale or to small experimental catchments where sufficient monitoring data are available (see *Pandey et al.*, 2008; *Pieri et al.*, 2007; *Schröder*, 2000 for examples of applications).

Only few models are found in the literature that are both physically-explicit and sufficiently efficient computationally to allow their application to larger catchments.

One recent example is tRIBS-Erosion [*Francipane et al.*, 2012]. This model integrates a geomorphic component into a hydrological model, thus explicitly accounting for feedbacks between erosion processes and the evolving landscape form. This coupling between processes makes the model more suitable to study river catchments as a whole; however, at the same time it limits the computational efficiency of the model.

SWAT is a daily time-step, semi-distributed model based on Hydrological Response Units (HRUs) suitable to reproduce continuous-time landscape processes at the catchment scale [*Neitsch et al.*, 2011]. It accounts for soil erosion on hillslopes by application of the Modified Universal Soil Loss Equation (MUSLE) and was also applied to large scale catchments such as the Blue Nile in Ethiopia [*Betrie et al.*, 2011]. SWAT has, however, some of the above-mentioned limitations, such as high input data requirements, and especially is missing spatial connectivity from sediment sources to sinks, due to the independence of the HRUs from the landscape [*Krysanova and Arnold*, 2008].

2.2 SOLUTE TRANSPORT IMPLEMENTATION

The implementation of solute transport modelling depends on whether the solutes are conservative or non-conservative. Distributed modelling of non-conservative transport processes is far more

challenging, and remains an open problem at the scale of medium to large catchments. The different implementations are described in sections 2.2.1 and 2.2.2 following.

2.2.1 Conservative solute transport

The concept of our study is the implementation in a fully distributed hydrologic model of a method of tracking conservative solutes and water age of non-conservative solute transport, and then using C-Q relations as a proxy for validating the model (see section 2.2.2 for more detail of the C-Q relations). This solution allows to keep the model flexible enough for use with variable amounts of data available for calibration, while focusing on the detailed simulation of the transporting agent, water, through a purely advective mechanism. Introducing a detailed small-scale representation of the non-conservative processes in each of the physical compartments where they occur, would automatically limit, under the present conditions of knowledge and computer power, the ambit of use of the model to small scale experimental basins, due to the high computational demand and data requirement. We believe that tracking correctly water parcels across the catchment compartments and combining this knowledge with robust conceptual relationships allows obtaining a distributed description of solute dynamics, which can be conveniently used to understand the risks associated with agricultural, domestic and industrial pollution, provided that the source location and the solute injection amount are known. The latter is, in the largest part of the cases, still very problematic, despite the on-going effort to improve monitoring.

To develop the hydrological and conservative-solute tracking component of TOPKAPI-ETH we rely on the Water Age and Tracer Efficient Tracking (WATET) model [*Remondi et al.*, 2018] as a blueprint. While keeping the essential components of a process-based hydrological model like TOPKAPI-ETH, WATET remains relatively efficient in terms of computational time for catchment-scale, long-term, and high-resolution distributed simulations and, thus, represents a convenient candidate solution for integration into TOPKAPI-ETH and further development towards a non-conservative model.

WATET includes a regular gridded representation of the catchment and computes the routing from cell to cell at each time-step. The outputs of the simulations are computed at the hourly scale, while the internal time step is at higher resolution. The model tracks conservative solutes through the catchment, thus allowing the *a posteriori* computation of the TTDs. In order to characterise water transit and residence time distributions, WATET tracks water and solutes from individual precipitation events, and solutes falling on different parts of the catchment. The hydrological model is therefore coupled with a module simulating solute transport and water ageing.

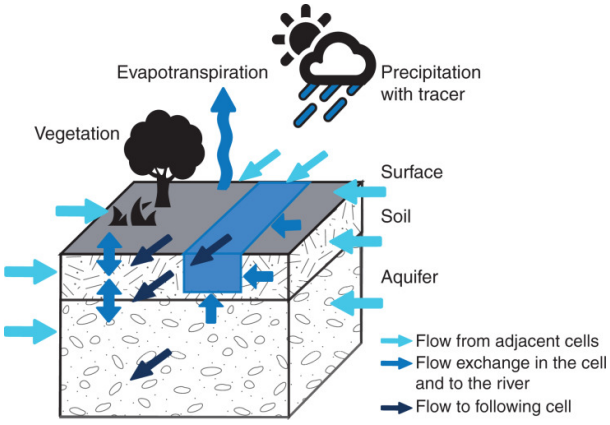

Figure 1. Schematic representation of WATET model [from *Remondi et al.*, 2018].

The hydrological component of WATET simulates water flow between and within cells on the surface, in the channel, and in the soil and aquifer layers, which mimic shallow and deep water storage (Figure 1). Each cell is connected to the surrounding ones in the surface and subsurface along topographic gradients. After accounting for precipitation, actual evapotranspiration (limited by available soil moisture) and infiltration, WATET simulates overland and channel flow using a kinematic wave approximation that accounts for surface roughness and follows the local topographic slope as it is done in TOPKAPI and TOPKAPI-ETH [*Ciarapica and Todini*, 2002; *Fatichi et al.*, 2015]. Overland flow can be generated by saturation excess and infiltration excess runoff. In the model, the soil water storage in each cell is recharged by infiltration, the rate of which is assumed to be the minimum between the precipitation rate and the soil saturated hydraulic conductivity. Soil water flow in the horizontal and vertical directions is controlled by hydraulic conductivity, whose dependence on saturation state is parameterized with van Genuchten conductivity functions [*van Genuchten*, 1980]. Saturated cells can feed the surface flow. Subsurface lateral flow is modelled by the kinematic wave equation, after vertical deep leakage towards the aquifer is computed. Groundwater storage is schematized as a non-linear reservoir equivalently to *Benettin et al.* (2015): each cell drains to its adjacent downslope groundwater storage or streamflow cells at a rate that is a power function of its local groundwater storage. The non-linear groundwater drainage function is assumed to be the same at every grid cell. If maximum groundwater storage is exceeded, water is transferred to the soil and potentially becomes saturation excess runoff. In order to improve the computational performance, water dynamics are solved with a 5-minute simulation time step, and internal time steps for surface overland and channel flow routing are set to 20 and 6 seconds, respectively.

The transport component of WATET explicitly calculates the spatially distributed water age and conservative tracer concentrations in the soil, aquifer and channels. The passive tracer can be introduced to the system with precipitation input and it can assume different concentrations in each cell and storage compartment (channel, soil, aquifer) at each time step. Moreover, dry deposition and evapoconcentration (the concentration of dissolved solutes at an evaporating surface) can be explicitly simulated in WATET. The model assumes that the conservative tracer follows the water (i.e., a purely advective behaviour) and that each storage compartment in each cell is well-mixed, without the presence of a residual or passive storage (e.g., *Hrachowitz et al.*, 2013; *Kirchner et al.*, 2010). Flow is non-age-selective, so the discharge from each storage compartment has the same mean age and tracer concentration as the water in that compartment during that time step. Tracer concentrations and mean ages differ among layers and cells, thus allowing catchment structure and geomorphology to play a major role in the tracer dynamics of the entire catchment, and to act as a distributed selection function.

The required input variables for the model are the distributed fields of precipitation rates and tracer concentrations, and potential evapotranspiration (PET) at each time step. PET depends on land cover, wind speed, air temperature, solar radiation, and air humidity. It is calculated externally and it is used as PET input to WATET. Generally, chloride is used as conservative tracer, because it does not react and it is sourced mainly by deposition, which, with the most innovative technologies, can even be sampled continuously.

2.2.2 Non-conservative solute transport

In order to investigate the impacts of agriculture on water quality in DAFNE, we are mainly interested in modelling non-conservative solutes, with a special focus on the main by-products of agricultural practices, e.g., nitrogen and phosphorus, in order to investigate the impacts of agriculture on water quality, it is necessary to modify the basic WATET model structure described above to include the dynamics of non-conservative solutes. Following the argument expressed above about the need for a solution that allows limiting the computational demand and simulating also basin scale transport, we opted for including a conceptual first order degradation kinetic, which does not focus on the detailed processes description but on a lumped evaluation of the amount of degraded solute during the export across the catchment, but still tracking such evaluation in a distributed

fashion across the catchment and across the different model compartments (surface, soil, groundwater storage, channel, …).

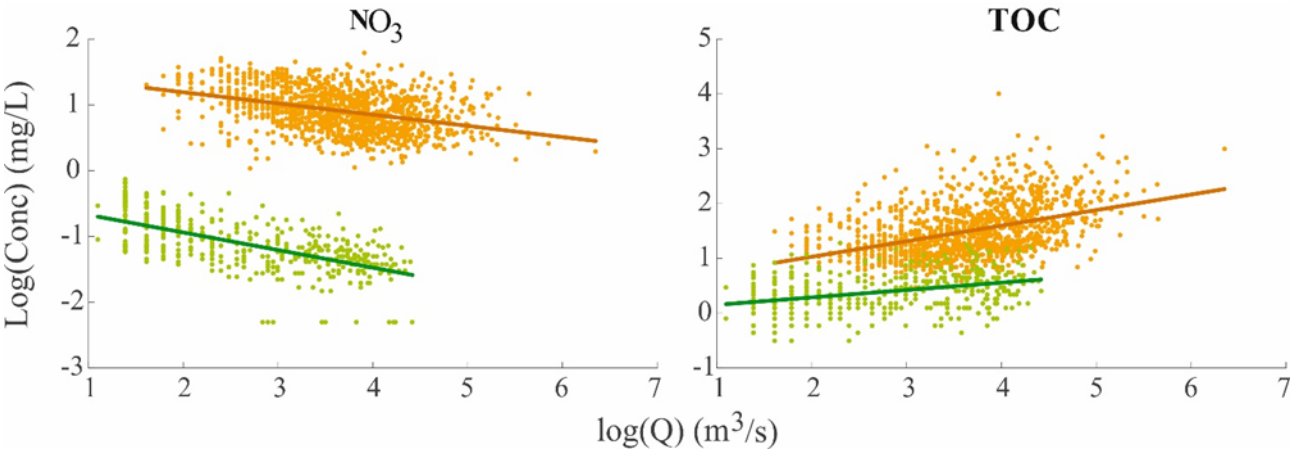

Figure 2. Typical examples of C-Q relations. The C-Q relations refer to two Swiss catchments, the Thur (orange) and the Inn (green). The first one is located in northern Switzerland in the Swiss Plateau area, while the second one is in the Alpine region in southern Switzerland. The left panel shows C-Q relations for nitrate (NO_3) while the right panel for total organic carbon (TOC). For NO_3 the difference between the two curves is bigger than for TOC, since the Thur catchment is mainly agricultural and therefore the input of fertilisers (containing NO_3) is much higher than the input in Inn catchment, the mainly forested catchment. TOC, instead, is related to the presence of sediments and particles entrained from soils. Erosion is high at steeper morphologies, like the Alpine one as well as in agricultural lands and therefore the difference between the two C-Q relations is not big. Moreover, with higher discharge the erosion is higher, resulting in a positive slope of the C-Q relation [modified from *Botter et al.*, in review].

The first order kinetic expression relates the concentration of the solute at each time step in each cell of the domain to the residence time by means of a degradation constant k_{deg}, which is the only parameter requiring calibration. The expression is formulated as:

$$C(t) = C_0 e^{-k_{deg} t} \quad (1)$$

Where C_0 is the initial concentration in each cell of the catchment and t is the residence time of the given solute in the cell. The concentration is computed for each time step in each cell of the domain and the parameter k_{deg} assumes different values in the different compartments of the catchments (i.e., surface, soil, groundwater and channel), reflecting the compartment-specific dynamics.

We aim at a para-calibration using *C-Q* relations as descriptors of solute export dynamics at the catchment scale. Concretely, given a known catchment configuration and hydrological dynamic validated on the discharge data (or also other data, if available), we aim at finding the non-conservative model parameter sets that best represents the *C-Q* relations at the outlet, two examples of which are shown in Figure 2.

While the *C-Q* relations alone have per se a limited predictive power, when integrated into a spatially and temporally explicit modelling framework of water fluxes across the basin compartments they can provide valuable information on the non-conservative solute dynamics in the catchment, which, in turn, provides interesting insights into the consequences of specific development pathways and their combination with scenarios.

C-Q relations have to be inferred from joint observations of flow and concentration for each of the considered solutes. This enables not only the evaluation of the specific solute behaviour, but also, by comparing different solutes, the inference of the relation between typical solute characteristics and the observed behaviour. This allows to isolate specific dynamics for specific solutes, thus be-

ing a proxy for more complex models, which are suitable only for small scale applications. Moreover, if data for a specific solute are available from different (sub-)catchments, their comparison enriches the analysis by allowing the analysis of the impacts of different catchment characteristics on the solute dynamics.

The suggested framework has already been applied to pilot-case studies and the reader can refer to *Remondi et al.* (2018) concerning the hydrological and conservative solute tracking part, while to *Botter et al.* (in review) for the *C-Q* relations analysis.

To our knowledge and based on information collected during a visit to Zambia in July 2018, the Zambezi River Authority (ZRA) is the only agency monitoring the water quality in the ZRB on a regular basis, since the end of 1990s. Indeed, so far as we can determine, no national standards concerning water quality exist in Zambia, with the exception of a few guidelines that set limits on certain pollutant concentrations. Consequently, there is no legislation to support spending on regular monitoring and the existing water quality data are therefore fragmented and are often related to spot measuring campaigns carried out by single groups for a specific purpose. The interaction between the Partners (UNZA and ETHZ specifically) and ZRA was fruitful in terms of data sharing. In July 2018 an agreement between the Partners and ZRA was found and the ZRA water quality and flow data were shared. For a more extensive description of the selected input data for the water quality model the reader is addressed to Milestone 7[1]. Figure 3 shows some preliminary results: the *C-Q* relations computed with the ZRA data at the Victoria Falls measuring station, located on the main Zambezi. The water quality analyses are carried out monthly and the data are available for the period 2010-2018. We analysed the concentrations of: dissolved oxygen (DO), total suspended solids (TSS), total dissolved solids (TDS), total phosphorus (TP), ammonia and chloride (Cl).

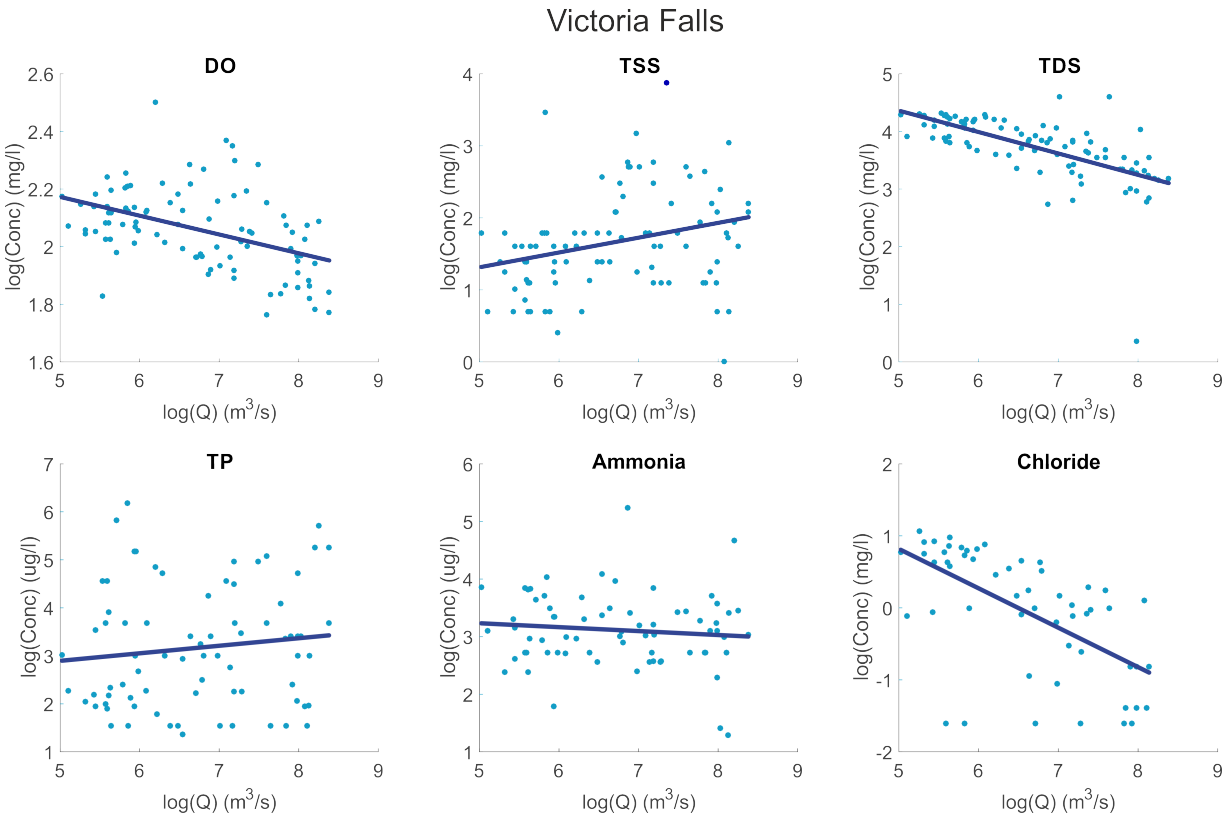

Figure 3: C-Q relations at Victoria Falls measuring station on the main Zambezi.

[1] In draft (to be uploaded to DAFNE Polybox)

Table 1 reports the values of the b exponents of the C-Q relations of Figure 3. The significance of the *b* exponent is tested with a Student's t-test and the resulting *p*-value is reported in Table 1. The threshold α for the significance test is fixed at 0.05, and we are testing the hypothesis that the exponent is significant in describing the relationship between log(C) and log(Q). Subsequently, if the p-value is lower than this threshold we consider the slope of the C-Q relation as significant and the behaviour is determined by the sign of the b exponent. If the relation is not significant the slope of the C-Q relation is considered to be near-zero and the behaviour is considered to be bio-geochemically stationary.

Table 1: Results of the C-Q relations analysis at Victoria Falls.

Solute	***b***	***p* value**	**Behaviour**
DO	-0.065	0.000	Dilution
TSS	0.206	0.007	Enrichment
TDS	-0.372	0.000	Dilution
TP	0.157	0.297	Bio-geochemically stationary
Ammonia	-0.068	0.398	Bio-geochemically stationary
Chloride	-0.547	0.000	Dilution

Concerning the OTB case study, it is still unclear if any long-term water quality monitoring programs exist, but from our preliminary research it appears unlikely that such data are available.

It must be finally observed that, even in the case flow and concentration data are available to infer the C-Q relationships, the main issue related to the non-conservative solute transport modelling concerns the availability of spatially distributed time series of concentration at the source, be that the area of application of agrochemicals, or the outlet of sewage systems or industrial settlements. The sources of these solutes are mainly non-point sources of different origin (e.g. field fertilisation, waste water treatment plants, industry, …) and usually no records of the solute inputs into the basin exist. This makes the validation of a model to predict the concentrations of these solutes at the river outlet hardly possible, unless plausible assumptions are made on the input concentrations for the most relevant solutes on the basis of proxy information. Considering the specific cases of the ZRB and OTB case studies, for example, in the best scenario, annual fertiliser loads per country might be available, but the downscaling in time and space of this input requires additional information on the type of product and its typical application scheme likely used as a function of the crop and/or agricultural practice. As a consequence, this may lead to limitations when a rigorous calibration and validation of the model is attempted.

2.3 SEDIMENT TRANSPORT IMPLEMENTATION

The dynamics of several compounds and/or elements follows strictly that of sediments, as they are adsorbed to soil particles. Therefore, in order to simulate their transport (e.g. in the case of phosphorous and its compounds), it is necessary to simulate properly erosion and sediment transport processes. The approach, which have implemented in the hydrological model is described hereafter.

The sediment module of the coupled hydrological-water quality model computes sediment production through soil erosion and mobilization of deposited sediments on hillslopes and subsequently routes them from the hillslopes, through the channel network to the catchment outlet.

Sediments are produced on the hillslopes via overland flow erosion according to one of three different approaches, to allow for a broader range of reproduced erosion mechanisms, as outlined in equations 1-3 below. First, soil detachment in each cell of the distributed model can be assumed to

be equal to the transport capacity, computed according to the formula proposed by *Prosser and Rustomij* (2000):

$$T_{C\_E} = \alpha\, S^{\beta} q^{\gamma} \quad (2)$$

where S is the cell slope, q the specific water discharge and α, β and γ are calibration parameters. Alternatively, soil detachment can be computed as a function of the shear stress at the interface between overland flow and soil surface [*Kilinc*, 1972], or of the stream power [*Elliot and Laflen*, 1993], respectively expressed by equations (3) and (4):

$$q_{sedE} = k_t(\tau - \tau_{cr})^{\mu} \text{ with } \tau = \rho\, g\, h\, S \quad (3)$$

$$q_{sedE} = k_{\omega}(\omega - \omega_{cr}) \text{ with } \omega = \rho\, g\, q\, S \quad (4)$$

where ρ, g and h are water density, gravity and depth respectively, while k_t, k_t, τ_{cr}, τ_{cr} and μ are calibration parameters.

If the sum of the mass of soil detached in the considered cell and the input of sediments from the upstream cells exceeds the transport capacity, the sediments in excess will be deposited in the current cell and the outflowing sediment discharge will be equal to the transport capacity. In the opposite case, the outflowing sediment discharge will remain equal to that computed by equation 1, 2 or 3 and the eroded sediments taken from the upper and lower soil layers.

The detached sediments are then routed from their sources on the hillslopes until the channel network, where they are transported via suspended sediment transport or bed load sediment transport depending on their grain size.

The process of suspended sediment transport is usually described with an advection-diffusion equation; however, in the TOPKAPI-ETH model, the process of diffusion has been neglected in order to be able to solve the equation analytically (as it is for most of the process components simulated by the model) and thus keeping the computational demand consistent with that necessary for the other modelled processes. The transport equation for fine sediments therefore reduces to the 1DV advection equation:

$$\frac{\partial AC}{\partial t} + \frac{\partial QC}{\partial x} - S_{bed} = 0 \quad (5)$$

where A is the area of the river cross section, C the fine sediment concentration average on the cross section, Q the river discharge and S_{bed} the term of sediment exchange with the bed.

The transport of the coarser sediments instead, takes place via bed load transport. Several formulas have been implemented and tested in the model for the computation of the bed load transport capacity, in order to allow for a broad range of transport conditions. These include *Meyer-Peter and Müller* (1948), *Recking* (2010), *Wilcock and Crowe* (2006) and two formulations for steep channels by *Rickenmann* (1990, 2001). An initial bed load transport formulation had been implemented by *Konz et al.* (2011), but this has been overhauled, and the additional formulations developed under DAFNE.

In the absence of data from the case studies, the model was so far developed and partially calibrated on the Kleine Emme river basin, located in Switzerland. This basin was chosen as a reference case study because measurements of the suspended sediment concentration are available at the outlet and an estimate of the bedload transport can be found in the literature. It is important to gain some confidence in the model before moving to poorly monitored catchments (we have recently started work on a case study in a sub-catchment of the OTB, see section 0).

2.3.1 Hillslope erosion

An example of the potential of the erosion and transport model on hillslopes is provided in Figure 4, which shows the map of the soil thickness at the beginning of the simulation and after 1.5 years of simulation on the Kleine Emme catchment used for testing the model. The figure shows how the simulation is able to reproduce hillslope erosion that is coherent with topographic features. Indeed,

erosion is concentrated in gullies and secondary channels, which are not defined as part of the modelled river network, but are classified as hillslopes in the flow accumulation map of the model terrain description.

Sediments eroded on the hillslopes are then routed in the river via suspended transport, thus allowing, in the hypothesis of neutral erosion/deposition in the channel, the validation of the magnitude of the erosion and deposition on hillslopes through the comparison of measured and simulated suspended concentrations at the outlet.

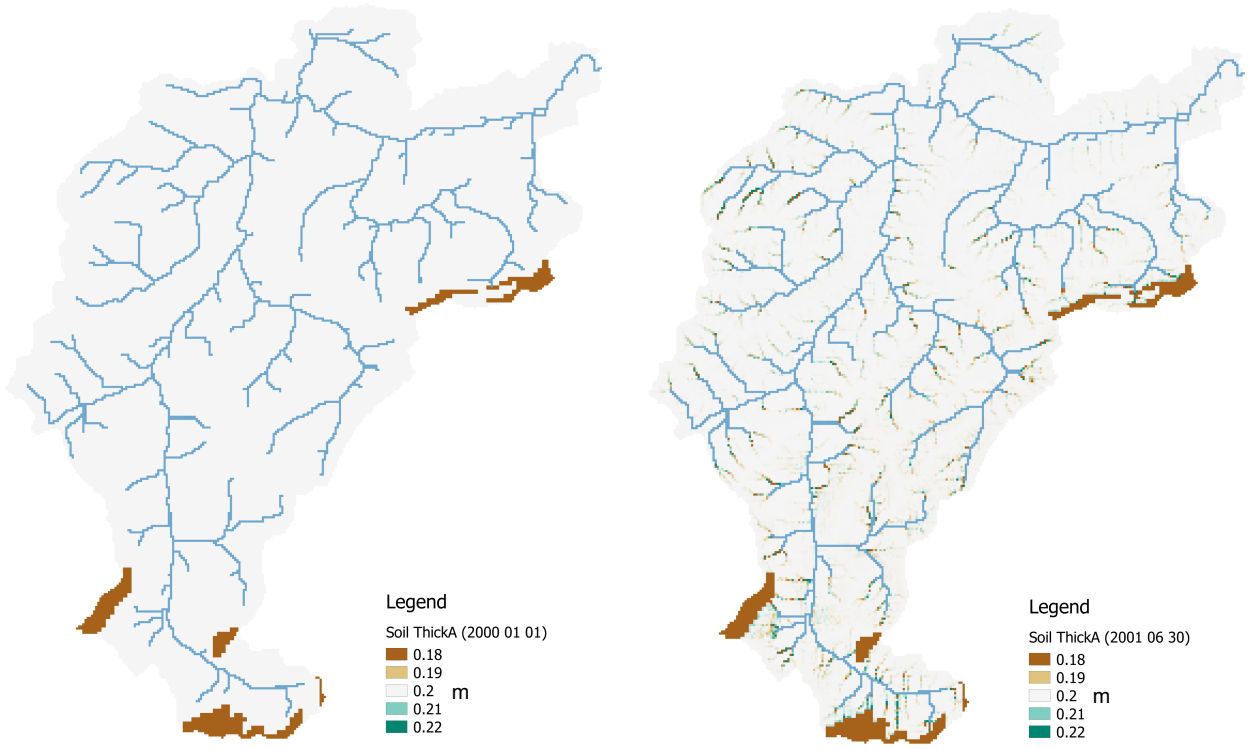

Figure 4 Initial (left) and final (right) sediment thickness in the Kleine Emme basin (477 km^2) after 1.5 years of simulation. This simulation was carried out with an artificially constant initial sediment thickness of 0.2 m over the entire catchment. Therefore, the areas coloured brown in the right-hand panel represent sites with erosion, while the green colours represent deposition. The areas with 0.18 m thickness, which appear to remain unchanged between the start and end, are rocky areas, where soil is thinner. Due to their low erodibility, the small changes in these areas cannot be represented in this colour scale.

2.3.2 Bed load transport

The test simulations provided good results also for the bedload sediment transport. Figure 5A shows the estimate of bed load transport for the lower part of the main channel of the Kleine Emme, proposed by *Heimann et al.*, (2013). The estimate is given in terms of accumulated bed load transport for the period 2000-2005, i.e. the cumulative sum of sediment volume that has flowed through the considered cross sections during the period 2000-2005.

Figure 5B shows the output of TOPKAPI-ETH simulation for the same period. One can observe that the model correctly reproduces the increasing trend of the accumulated bed load transport in the downstream direction, and captures the localised inputs of sediments from the two main tributaries.

A reasonable agreement is found not only in terms of temporal and spatial dynamics but also between the observed and simulated volume of transported sediments at the outlet, as both values are in the range 1.6 – 1.8·10^5 m^3. The simulated mean yearly volume of transported sediments is

$2.8 \cdot 10^4$ m^3, which also compares well with the observed value of $3.8 \cdot 10^4$ m^3 (see *Heimann et al.*, 2013, and *Hinderer et al.*, 2013).

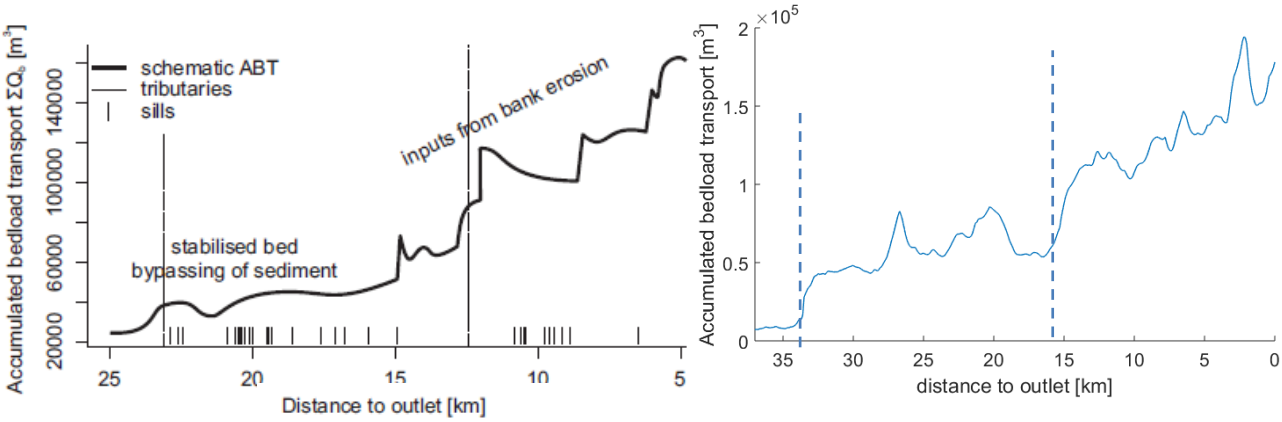

Figure 5 Accumulated bed load transport along the final part of the Kleine Emme main channel for the period 2000-2005. On the left is an estimate by *Heimann et al.*, (2013) and on the right the values obtained with TOPKAPI-ETH model.

2.3.3 Suspended sediment transport

The suspended sediment transport (SST) module has been implemented in the model as described in §2.3, but the calibration on the data is still in progress. More features are also expected to be added to the SST module to improve the representativeness of the model. These include the separation of sediments eroded on the hillslopes into suspended and bedload transport, and the implementation of fine sediment storage in the channel bed.

Preliminary results are shown in Figure 6, which presents the modelled and measured discharge at the outlet, together with the concentration of suspended sediments in the flow and the mass flux of suspended sediments carried by the river.

We can observe a direct correlation between the simulated discharge and the sediment flux at outlet, while the trend of suspended sediment concentration differs from the one of the flow. This difference can be explained by the spatially distributed sediment production process implemented in the model, as opposed to an approach using a discharge to suspended sediment concentration (SSC) relationship. With the implemented approach, rainfall events resulting in a small discharge at the outlet can be characterized by higher SSC than found during high-discharge events, when the first are generated by localized rainfall events in sediment rich source areas, while the second result from wide-spread rainfall causing homogeneously distributed overland flow to deliver the same (or less sediment) in a greater volume of water. This dichotomy is evident for the events of 24 Apr-9 May 2001 (low discharge, high SSC – generated by a localized event) and 16 July (high discharge, relatively lower SSC – spatially distributed overland flow). Figure 7 illustrates the source of flow and sediments for each event, by plotting a map of the peak overland flow rates for the events.

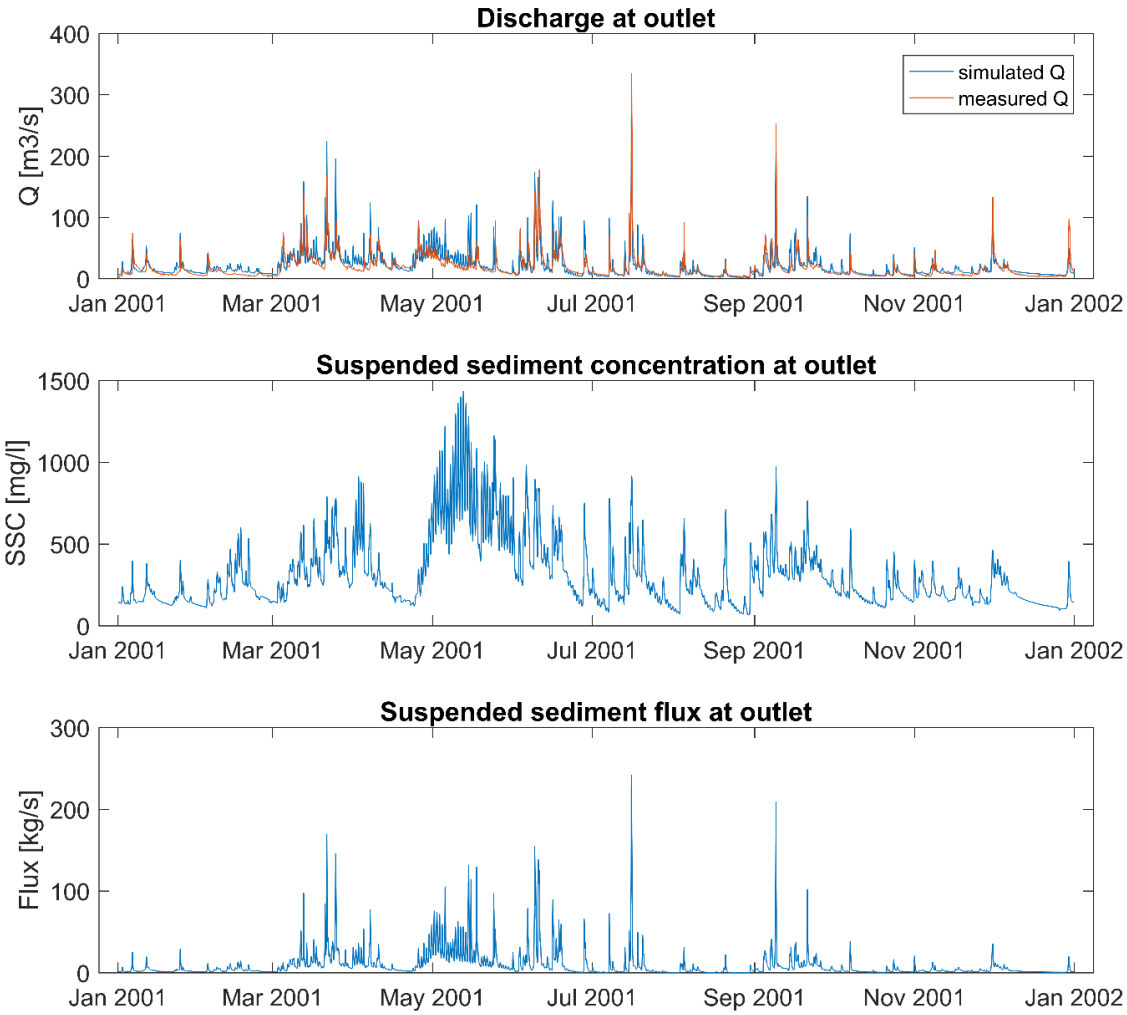

Figure 6 Time series of modelled and measured discharge (top panel), together with the concentration (middle) and mass flux (bottom) of suspended sediments at the Kleine Emme river outlet for a one-year simulation. Notice the strong correlation between modelled flow and sediment flux, which is not evident in the sediment concentrations – this is a result of the spatially distributed sediment generation process in the model.

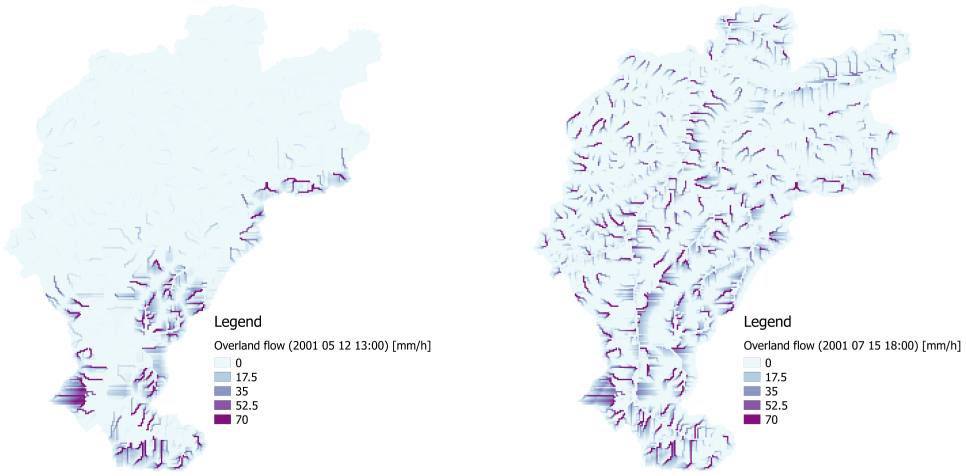

Figure 7 Maps illustrating the source of flow and sediments for the events of May and July 2001 (refer to time-series in Figure 6), by plotting a map of the peak overland flow rates for the events.

The annual suspended sediment yield at the outlet produced by the model is of $1.44 \cdot 10^5$ t, while the yield obtained from measurements is about $2.83 \cdot 10^5$ t. While the order of magnitude of the suspended sediment yield is already captured correctly, there is need for additional work on the calibration of the model, particularly on the bed sediment exchange term, for which observation are hardly available anywhere.

2.3.4 Preliminary work on hillslope erosion investigation in the Gununo catchment, OTB, Ethiopia

As a first effort towards using the model in the DAFNE case studies, we are working on a hillslope erosion study in the Gununo site in the north-eastern part of the Omo river catchment.

In the context of the Soil Conservation Research Program (SCRP) carried out by the University of Bern from 1987 to 1991, runoff and soil erosion measurements were taken in a 1.69 km^2 area of the Omo river basin. The area is small, but representative of the high-potential perennial crops growing in the agro-ecological region in the southern and southwestern highlands of Ethiopia [*Tegene*, 1992).

The study area belongs to the Gununo soil conservation research site (Wolayta Awraja, North Omo Administrative Region) and is located at 37° 38' E and 6° 56' N, spanning an elevation range of 1900 – 2100 m asl (see Figure 8). The area comprises two small catchments with areas of 0.74 km^2 and 0.95 km^2, where the first one was subject to soil conservation measurements (parallel level bunds), while in the second one traditional land use was preserved in order to provide a basis for comparison [*von Gunten*, 1993].

We aim to test the TOPKAPI-ETH sediment transport module in the Gununo catchments by making use of the measured and observed quantities provided in the SCRP reports.

Input data

Climatic data were collected at the location during the project; in particular, rainfall amount and intensity is available from 1.1.1982 to 29.5.2000 on an event basis and the maximum and minimum air and soil temperatures are available for the period 1.1.1986 to 30.9.2001. Evaporation has also been measured by means of an evaporimeter from 1981 to 1988.

Land use is described in the map of 1988 (Figure 9), which shows 50% of the area under cultivation, 13% covered by bush and forest and the remaining 37% by grassland. The cultivated area is characterized by small-scale farms, with an area of approximately 0.5 ha per farm.

The soils of the area have also been characterized in the context of the SCRP and are described in detail in the Research Report 8 [*Weigel*, 1986].

The predominant soil types found in the area are Nitosols and Acrisols; these soils are generally very deep and fertile. Nitosols are the predominant soil form and cover about two-thirds of the area; they are mainly found on sloping grounds, while the gentler slopes and flat areas are characterized by Acrisols.

Calibration data

River discharge was monitored at the catchment outlet, together with the sediment yield, for the duration of the monitoring program. Sediment concentrations were measured by collecting 1L samples every 10 minutes during storm events. These data are currently not available at the time of preparing this report.

Additional soil loss measurements were taken around the catchments at different spatial scales and represent the effect of scale-dependent erosion processes:

- 6 Micro-plots of 1x3 m measured rainsplash and sheet erosion in interrill areas;
- 8 Test-plots of 15x2 m also included the processes of pre-rill erosion and diffuse accumulation;

- 6 Experimental plots of 30x6 m include the effects of rill erosion, diffuse and concentrated accumulations. These plots represent a sequence of terraces and soil conservation structures interrupting runoff and soil transport.

At the scale of the entire study area, mapping of rills and gullies created during storm events was also performed and collected in damage maps.

This description of the Gununo sites has been included to direct the reader towards our plans for working towards developing the WEF model within the case study regions of DAFNE.

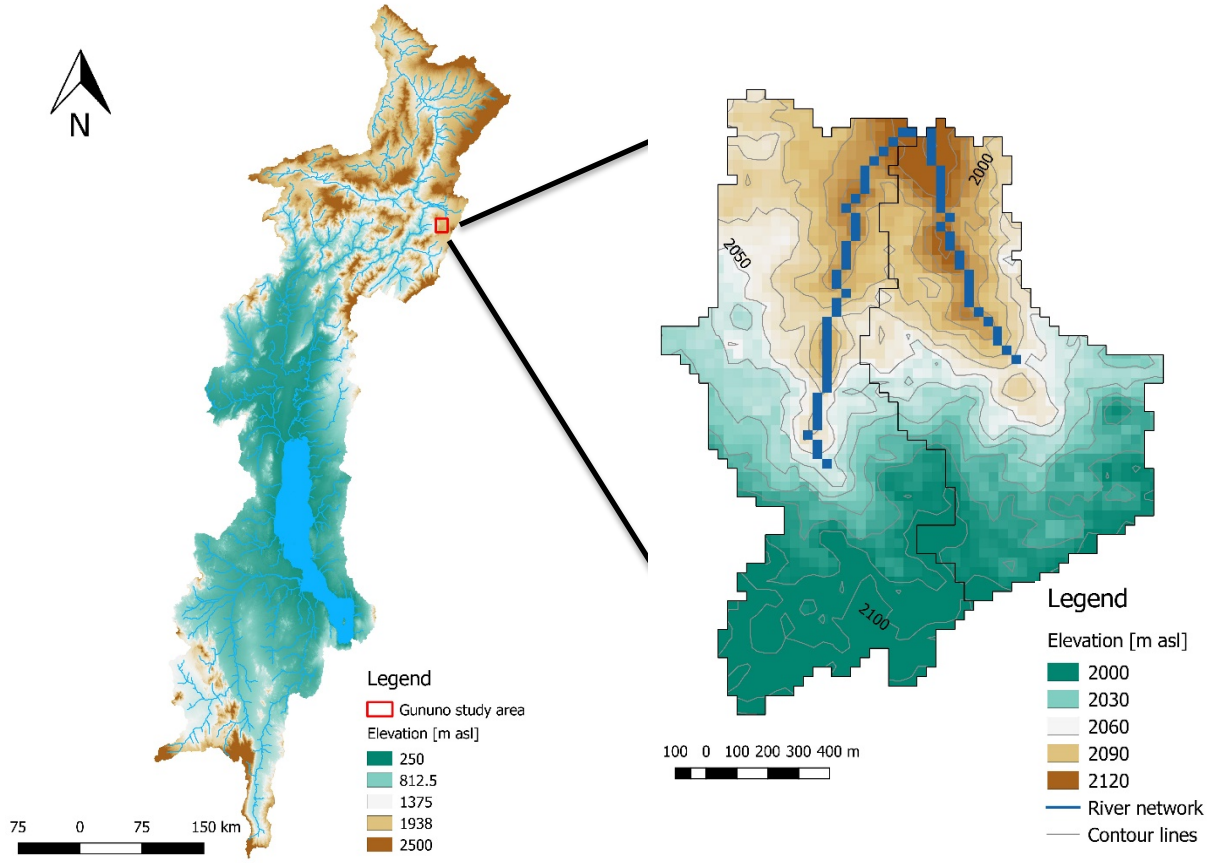

Figure 8 Omo-Turkana Basin with location of the Gununo study catchments.

3. TOPKAPI-ETH MODEL REDESIGN

Although the basic structure and concepts of the TOPKAPI-ETH model are retained, we embarked on a ground up reconstruction in order to escape from the accumulated "technical debt" (e.g. *Holvitie et al.*, 2018), which is the inevitable result of a model code developed during a long period of time (approximately 17 years) by numerous contributors with widely varying levels of programming experience and research interests. This does not imply that the previous contributors did poor work, only that as more changes were layered on top of each other, so it became more and more difficult to make the structural improvements to the code base required to develop the core engine of the integrated WEF model.

In this section we present an overview of the TOPKAPI-ETH model concept (i.e. algorithms and governing equations) and detail the structural and technical revisions, giving reasons for the choices we have made given the goals of the integrated WEF model and the available time and resources.

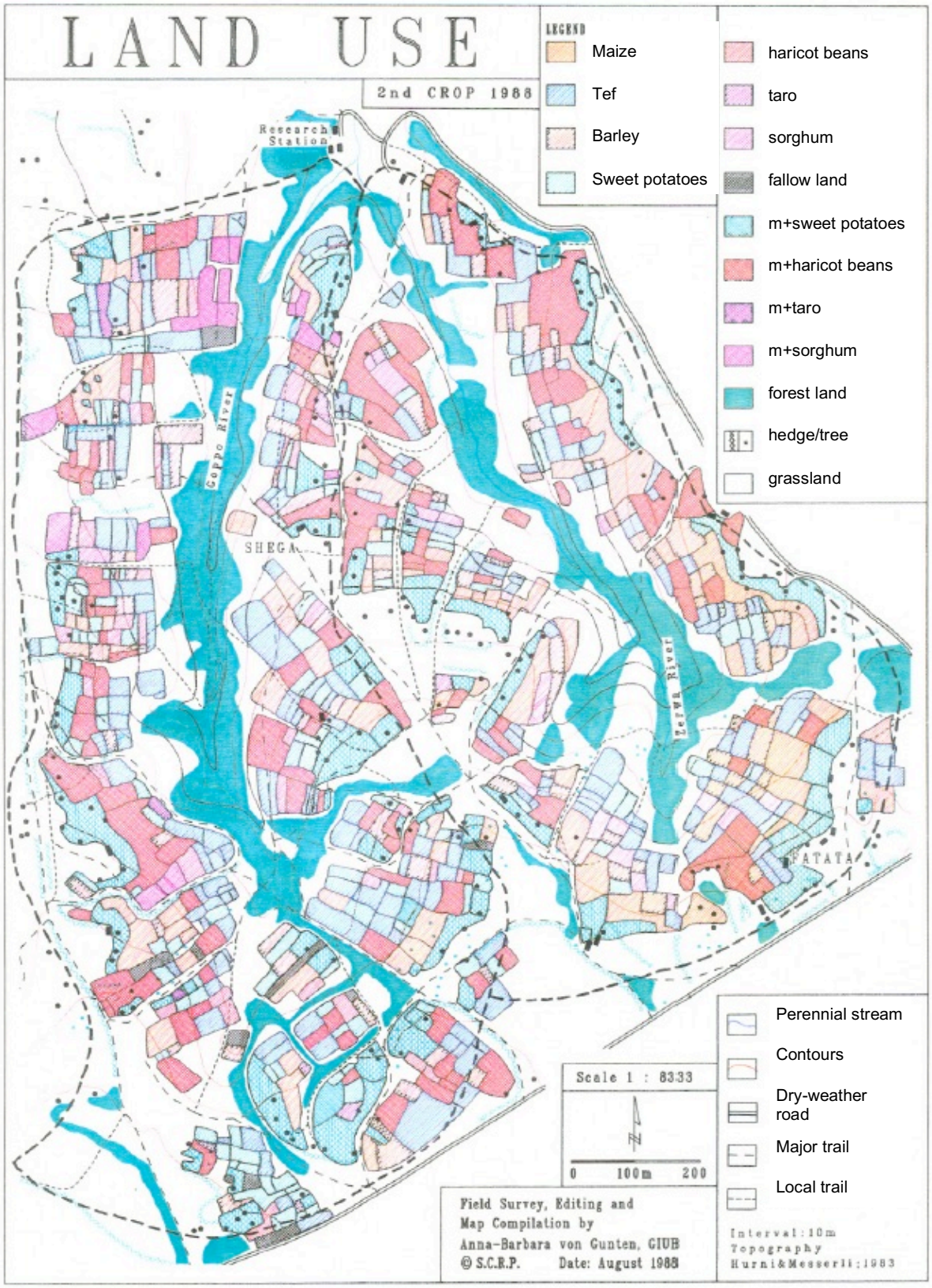

Figure 9 Land use map of 1988 for the Gununo basins, from *von Gunten* (1988).

3.1 MODEL DESCRIPTION

TOPKAPI-ETH is a comprehensive hydrological model first developed as a rainfall-runoff model at the University of Bologna by *Ciarapica and Todini* (2002) and subsequently extended by the Hydrology and Water Resources Management group of ETH Zurich [*Fatichi et al.*, 2013; 2015].

The TOPKAPI-ETH is a fully distributed, physically-explicit hydrological model. The catchment domain is discretized in the horizontal dimension by a raster map; the value of each hydrological variable is assumed to be constant within each cell.

The surface of each cell can be assumed to be covered by soil, water or snow. The subsurface of each cell is discretized in the vertical dimension as a column of three layers: the upper and lower soil layers, below which the groundwater layer is found (see Figure 10).

The two soil layers are schematized as non-linear reservoirs, while the groundwater layer as a linear reservoir useful to simulate slow flows.

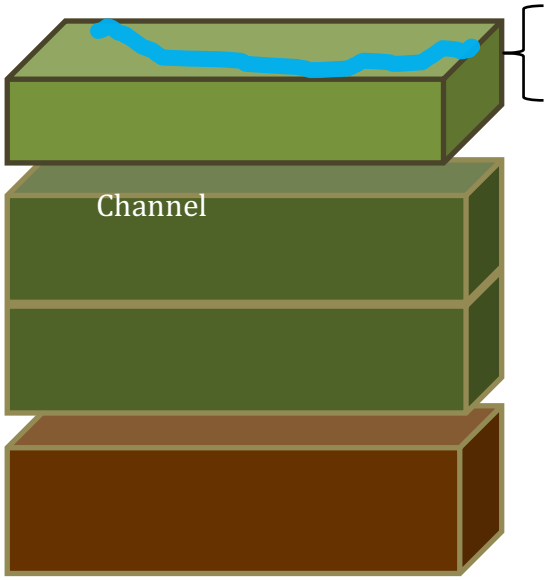

Figure 10 Sketch of the configuration of the various water stores in a TOPKAPI-ETH model cell.

In the current version of the model, the horizontal flow paths are determined based on a D4 routing scheme applied to the basin topography, which is described by a DEM. For each cell, there is a vertical connection between the compartments at all levels (exchanges between surface, soil and groundwater stores can take place).

The model needs meteorological forcing as input values, namely precipitation, cloud cover and temperature time series.

The user-input precipitation on the single cell is divided between interception and water that actually infiltrates the soil. Potential evapotranspiration is calculated using the Priestley-Taylor equation and limited by the actual soil moisture available in the first soil layer [*Priestley and Taylor*, 1972; *Brutsaert*, 2005]. Potential infiltration is regulated as a simple threshold function of the vertical saturated hydraulic conductivity or explicitly using Green-Ampt [*Green and Ampt*, 1911], and overland runoff can be generated by either infiltration or saturation excess.

A kinematic approximation is then used to route the topographically driven (Figure 11) subsurface water flows, surface overland flow and channel water [*Liu and Todini*, 2005]. The core concept of the model is that the kinematic approximation is framed as a non-linear Ordinary Differential Equation (ODE) with an identical form for each of the different kinds of store (soil, overland, channel),

and a pseudo-analytical solution is determined for each that only depends on the physically derived parameters for the cell. This avoids the use of a numerical solver under most conditions and makes the model computationally efficient, despite being physically-explicit.

Snow and ice melt are calculated with the empirical enhanced temperature index model, which requires air temperature, shortwave radiation and albedo only [*Pellicciotti et al.*, 2005; *Carenzo et al.*, 2009].

The model can also account for the presence of lakes or reservoirs within the basin, and can include a variety of water transfers between different parts of the catchment (for irrigation, or other consumptive use).

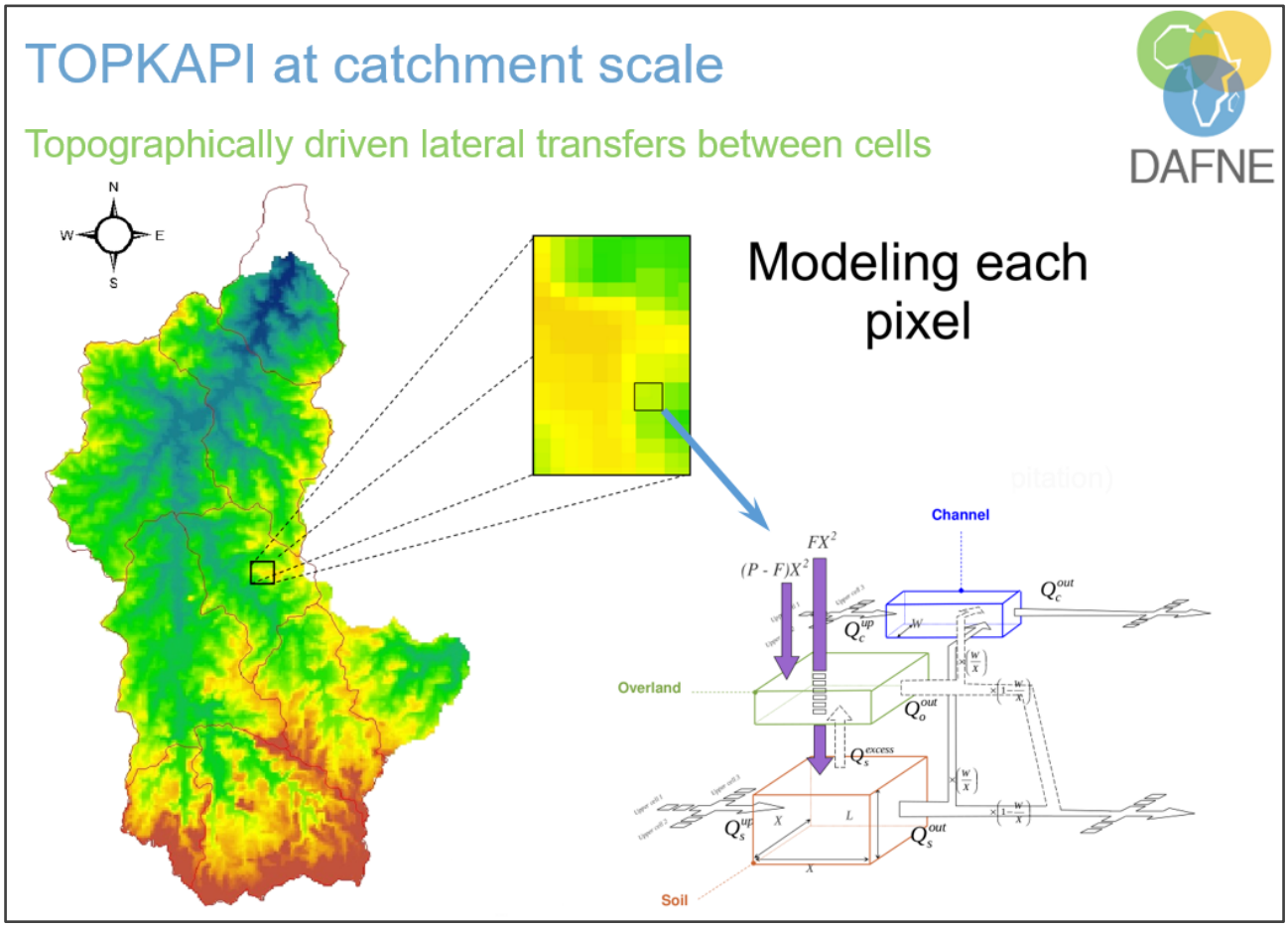

Figure 11 Illustration of the topographically driven structure of TOPKAPI-ETH – water flows from upstream to downstream cells within a catchment, until the outlet is reached. The interconnection between different water stores as shown in Figure 10 is also outlined.

3.2 CONCEPTUAL DESIGN

In this section we focus on aspects related directly to the coding and logic of the model implementation, while in section 3.3 we concentrate on issues related to the building and deployment of the model. To improve the readability of the report, we have tried to include the main technical points, while avoiding too much detail. Instead we have included such detail in Appendix A (section 7).

Based on the analysis of the TOPKAPI-ETH implementation in section 7, we have identified the main drawbacks of the current version of TOPKAPI-ETH as being related to problems that are typical of many scientific programs. These are: the programming and comment styles that describe the

logic and architectural flow of the program are non-homogeneous, which is caused by different authors in different years; there are numerous global variables without write protection, and these are typically passed as procedure arguments with R/W rights, thus breaking the best-practice of encapsulation and increasing the possibility of bugs that are hard to diagnose; there is no infrastructure to handle exceptions in I/O and run-time operations, so when these occur the result is an unhandled crash; the source files are organized into subfolders but should also be organized as Fortran modules to encourage code reuse and encapsulation; some structural hypotheses (e.g. the D4 horizontal connection between cells) are hidden and distributed (often hard coded), making them challenging to change without breaking the model; finally, some flow control tests (e.g. "if" statements) use direct equality between reals without considering floating point precision, a common usage pattern is when a "select case" statement uses real variables to switch between subcases, likely leading to bugs that are hard to resolve.

From the point of view of model performance, our runtime code profiling has shown that there are some bottlenecks in the current version of TOPKAPI-ETH (summarized in words here, for the numbers see appendix A Figure 18). As was expected, the main computational burden is in the solution of the non-linear reservoir equations, which are kinematic flow approximations used to propagate the flow in channel, surface and sub-surface (soil layers) components. Other non-negligible contributors to the calculation time are, the aggregation of information at the catchment scale (average, minimum and maximum values), the preparation and calculation of time-variable parameters for each time-step of the simulation, and especially the cost of I/O operations and the computation of global irradiance for the evapotranspiration estimation. In the redesigned code, we are systematically redesigning the logic to address these issues as far as possible.

The target for the new version of TOPKAPI-ETH is to implement a secure, fast, modularized and extensible code, which is designed to take advantage of a multi-core "traditional" server with no RAM limitations. At the same time, we aim to improve the overall usability of the model. The distinction between these two objectives is reflected in the different focus between the discussion in this section, and section 3.3.

To enhance security, in addition to systematically introducing generalized exception handling, some architectural choices were made to avoid global variables as far as possible; we also explicitly defined the intent of all procedure arguments (arguments are always defined as "in" or "out", with "inout" only used under exceptional circumstances); using "NaN" to initialize local variable and to substitute the "no data" code; introducing specific labels ("enum") to switch between subcases, instead of using comparisons between floating point numbers; sharing only necessary functions and procedures among modules, to ensure separation of the public interface from private implementation details; using as procedure arguments array defined with assumed-shape hypothesis, when possible; and, finally, creating systematically specific – small – procedures to implement automatic code testing for each modules functionalities.

In order to enhance the opportunities for increasing the computational performance of the code, we introduced the use of "elemental" procedures, to support parallelization. We also use generic programming constructs, to support downscaling from double to single precision or upscaling to quadruple precision, in a prototypical arrangement so that all the modules can be simulated in an adaptive way, specifying for each step the corresponding time interval (for example as a function of average conditions of the system state or of the mean variations of the forcing inputs). Other adaptations include the implementation of smart Euler solver of non-linear dynamic models with an adaptive internal time-step in solver process, the introduction of asynchronous writing operations and optimizing array access using the Fortran storage order and changing the smallest/fastest changing/innermost-loop index first. As explained in the next paragraph, the introduction of non-homogeneous spatial representation can be useful to produce a model of a very large system within reasonable simulation time.

Finally, to enhance flexibility and maintainability, the new version enforces a complete separation between the specialized calculous code and the flow of control in the program using formal Fortran

modules and an object-oriented programming paradigm. The naming of variables, structures, procedures and modules was made consistent throughout using a uniform convention. And the comments for procedures have also been written according to a template that will allow the generation of automatic API documentation (for example using the Doxygen[2] tool).

The implementation of the model code is structured as a collection of objects that are both independent Fortran classes and hierarchical sets of related classes. The objects are composed to build a catchment (system) model as described in the ensuing paragraphs, and illustrated in figures 12 and 13 below.

The basic element to build a system model is the class “Cell”, which is a single layer – rectangular element. This base class is responsible to set common properties like localization and geometry, and to store water volume, output flux and the details of upstream neighbours linked according to either a D4 or D8 scheme (user specified). The Cell class has a set of specializations which represent the different kinds of water stores present in TOPKAPI (soil, overland, groundwater, channel). The sub-classes “OverlandCell”, “SoilCell” and “GroundwaterCell” are stacked (individually or in layers) to represent the physical processes at increasing depth. The optional sub-class “RiverStretch”, is the basic element to reproduce the river network of the system. It is possible to specify different shapes for the river channel cross-section (current options are rectangular, triangular and trapezoidal). Each section definition within the network can be coupled with a different solver, which can be either the TOPKAPI kinematic flow approximation approach, or the Muskingum-Cunge-Todini (MCT) method (for cases of low slope when the kinematic approximation is not valid).

Other structural elements are “CellLoss” to represent a water withdrawal that will be transferred outside of the system, “RiverDiversion” and “WaterAbstraction” to reproduce a controlled water transfer respectively from point-to-point or point-to-area within the catchment (e.g. for irrigation, or other water transfer), and, finally, “Lake”, which represents one or more cells completely occupied by liquid water, and possibly regulated using different strategies (in the case of a hydropower reservoir for instance).

All these structural elements can be organized into logical computational groups. By default, this is as a “Stack”, which is a vertically composed collection of an “OverlandCell”, 1 or more “SoilCell” objects and 0, 1 or more “GroundwaterCell”. Each individual object is only having the responsibility to update its internal state and transform inputs into outputs that are available for other elements to consume. The “Stack” is responsible for the creation and linking of the basic elements (using the well know Factory Pattern[3] strategy), it also sets their properties, makes the necessary preparations for each simulation step and updates the elements of the collection in the correct order. An alternative structural organization of basic cells is the “Layer”, which represents a horizontally connected group of uniform cell types.

As the spatial scale increases, Stacks (or Layers) are organized in “Catch”, sub-basin elements of a water system model. In an analogous way to the Stack (Layer), the “Catch” is responsible for the creation and linking of it’s sub-groups (using the Factory Pattern strategy once again), and as before setting their properties, preparing for each simulation step and updating sub-groups in the correct order. “Catch” is responsible for storing the water table depth and running the simulation of large scale processes, such as the evapotranspiration process. Where possible, taking advantage of the elemental definition of calculation procedures.

Finally, the “Model” class is a Singleton element containing one or more “Catch” objects. It is the orchestrator of all the other structural and calculation modules except for the “Logger” (which independently writes information about simulation progress errors etc.). The “Model” controls the basic logic of the model, starting from parsing the command-line arguments, through configuration and

[2] http://www.doxygen.org/

[3] https://en.wikipedia.org/wiki/Factory_method_pattern

ingesting the input files, to finally control the simulation of global processes and the update procedures of all Catch objects.

Other relevant architecture design issues of this new version are:

- Definition of wrapper function to encapsulate method-specific calculation functions (using the Decorator Pattern strategy);
- Definition of utility containers (like vectors and queue) as to store time-variant parametrizations (for instance LAI parameter in the year, or land use along a multi-year simulation horizon, or minimum environmental flow from reservoirs);
- Extension of some parameters in space dimension (for instance to adapt lat/long/tz; or for some erosion parameters);
- Implementation of a “Reader” and a “Writer” Modules using the Singleton Pattern strategy. The first, to use different format for input files specified by templates or to possibly support remote reading; the second to support asynchronous writing, user-defined format by templates and eventually remote writing operation. To support eventually parallel simulation, a specific queue process has been implemented, to write consistent buffers as output;
- Implementation of a “Logger” (Singleton Pattern): to support hierarchical error messages generation on different streams. It is created directly from the main, to support possible problems in “Model” class creation;
- Implementation of a flexible configuration-file parser designed to support sub-grouped elements;
- Design of a “Linker (Singleton Pattern): to support logic provided by external services.

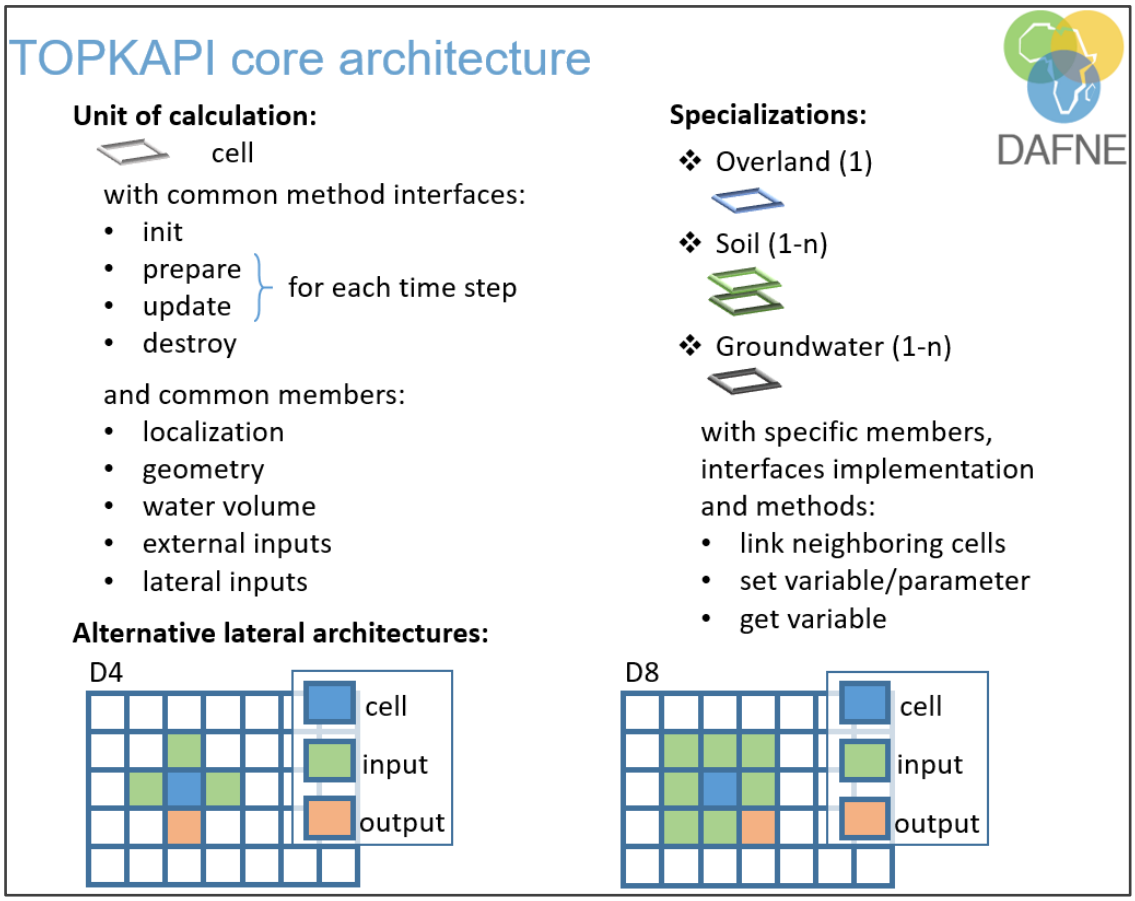

Figure 12. Concept sketch of model cell specializations and D8 flow paths

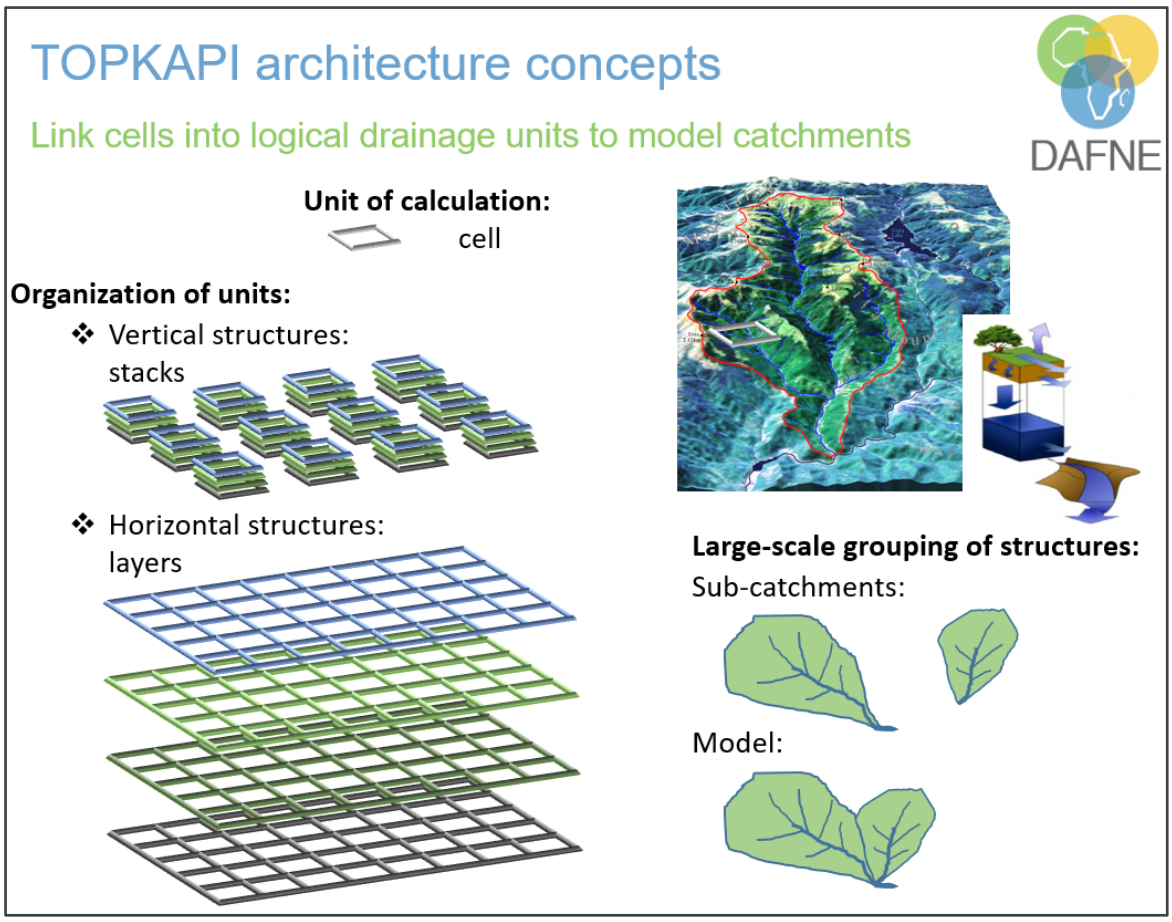

Figure 13. Concept sketch of model composition from cells to catchments

3.3 TECHNICAL ASPECTS

The existing version of TOPKAPI-ETH is a relatively complex research model built as a standalone single-threaded command line application available for the 64-bit Windows operating system (earlier versions ran on 32-bit Windows). As already discussed in section 3.2, one of the main motivations for the model redesign was to provide a more flexible architecture to (among other things) allow the model take advantage of modern multi-core processors performing calculations in parallel. This performance enhancement is necessary in the context of DAFNE in order to accommodate the large number of model runs required to evaluate multiple pathways within basins discretised into a large numbers of cells, and subsequently forced by ensembles of present and future climate scenarios.

Apart from the structural changes to the model design and coding constructs described in section 3.2, we have both implemented and planned a variety of enhancements in the way the model code base is managed, built and deployed. These enhancements and the integration of both new and existing pre and post-processing tools give the model flexibility to be managed and extended to a wider variety of new use cases. In this section we describe these technical aspects, which are rather different from those related to the model coding outline in section 3.2.

3.3.1 Source code version control

Typically, a model's source code is an evolving specification of the model's current structure and underlying algorithms. Version Control Systems (VCS) are a fundamental tool used by software developers to keep track of the evolution of a code base in a structured way. Although there are several variations of version control, the basic concept of the tool is to store uniquely identifiable

snapshots of the source code at different points in time. The advantage of this is that the VCS tools make it easy to see the exact changes between different versions and to therefore reproduce the exact model configuration used to produce simulation results (the value of reproducible results is explained in e.g. *Hutton et al.*, 2016). In addition, it is much easier to experiment with and test new model features without the risk of breaking working code (because it is easy to automatically undo the changes made using the VCS).

Due to the compelling reasons stated above, the TOPKAPI-ETH model code-base is now stored in a Distributed Version Control System (DVCS) making it simple to determine the exact version of the model used to produce a set of results and allowing us to track changes in the code and switch between different configurations in an objective and traceable manner. Collaborative development is also improved as individual developers can make independent changes that are later merged into a single code.

3.3.2 Build infrastructure and target operating systems

In order for the model source code to be built into an executable for a particular operating system it must first be compiled and linked using a compiler. The desire to run TOPKAPI on high performance clusters, as well as regular desktop machines, implies that it is necessary to build executables both for different operating systems, and with or without parallel computing infrastructure enabled. This very quickly results in the burden of maintaining several different methods of building the codebase for different target configurations.

To makes steps towards streamlining this complexity, the model's build infrastructure (i.e. tools to support the tasks described above) has received a significant non-trivial overhaul making it possible to now build the model using multiple different compilers, on several different operating systems using a single code-base (tested combinations so far mainly target the Linux and Windows operating systems using the GCC, Intel and Microsoft compiler tools). The single build infrastructure is based on the CMake[4] tool, which provides a single method to perform builds for multiple target configurations.

3.3.3 Model deployment

Model deployment refers to the method used to install the model and set up the model's runtime environment for a given computer (desktop, laptop), or high-performance cluster. Since we now target multiple operating environments this aspect becomes both more important and more challenging, and relies heavily on the work described in sections 3.3.1 and 3.3.2.

The existing method of model deployment is that the standalone model executable (Windows OS) is passed among students and researchers within the HWRM-ETHZ group, and several variations of the executable exist (depending on the exact source code used to build it). However, there is no structured way to be absolutely certain which variation of the model has been used to produce a set of results.

To date this aspect is in a state of flux, as we are experimenting with various alternatives. One option mimics the current arrangement with a stand-alone executable. The advancement is that we encode exact version information in the executable using the VCS tool, and we have the ability to build executables for different computing platforms using the build tools described in section 3.3.2. The second deployment option for which we already have a relatively advanced implementation is to deploy the model as a python library and command line executable – this is more flexible, but more challenging to code and build, so our progress is slower given our general lack of programmer resources.

So far, the TOPKAPI-ETH model core has been redesigned and implemented in Fortran as a compiled library of modular components, which can be used via an Application Programming Interface

[4] https://cmake.org

(API). The command line interface to the model is now driven from the Python programming language, and the basic support is in place to use the model as a Python library (e.g. from inside a GIS environment).

3.3.4 Integrated pre- and post-processing tools

We have incorporated the large library of existing pre- and post-processing tools written in MATLAB and have (so far) tested our ability to run these tools using Python as the driving language. There is still quite a significant amount of work to do to finalize the pre- and post-processing tools, however this is a low priority for DAFNE and will probably be deferred. Our main reason for bundling these tools into the TOPKAPI package is to make sure they are under version control, and that ongoing improvements and bug fixes are therefore handled in a structured and traceable manner.

3.3.5 Model input and output formats

The input and output file formats of TOPKAPI-ETH are both centred around formatted plain text files. While this has the advantage of being human readable and (in some cases) human editable, there are several opportunities to make improvements in this area, while still maintaining an approximate backwards compatibility with the (in)output files from existing simulation set-ups.

The files associated with TOPKAPI-ETH can be split into four groups a) configuration files describing the model set-up b) static parameter files that describe the model parameters (e.g. DEM, soil maps etc.) c) meteorological forcing time-series (gridded or station data) and d) various output files (time-series for grid cells, or spatial maps).

Since the configuration files are so specific to the model, there is not much benefit in making changes to their existing format and structure. For the other files, there are two main areas where beneficial changes can be made.

The first is to make sure that the model is capable of ingesting static parameter descriptions and meteorological forcings that are provided in well-known open-standard file formats. The advantage of this is that it gives the model user a much wider choice of software tools in which they can prepare the model input files. So far, TOPKAPI-ETH requires input map files in a standard ESRI Arc/Info ASCII Grid[5] (AAIGrid), but the meteorological time-series are an uncommon variant of Comma Separated Value (CSV) files. We have implemented a more flexible set of accepted inputs for the time-series that includes standard CSV files, and plan to increase the variety of accepted map inputs as future improvements.

The second area of beneficial change related to file formats, is to take advantage of standardized binary file formats for storing the model outputs. The advantage of this is that binary formats are much faster in respect of write speed, and also for file reads. In addition, the file formats typically used for data exchange in the science community are introspectable, meaning they have a typical file structure and enough meta-data that data users can discover the contents of the file without necessarily needing extensive documentation of the format – there are also many commercial and freely available tools available to read data-sets stored in such formats. Another advantage is that commonly used binary data exchange formats are portable among operating systems.

Based on our experience, we have chosen to implement support for CF (Climate and Forecast) metadata compliant[6] NetCDF format files for I/O operation in the new version of TOPKAPI. Unfortunately, official Fortran support[7] is a little weak at this point in time and we have not yet been able to use the official library. A work-around to solve this problem is to use the C version and implement an intermediate wrapper. At this time, this activity has low priority because we can use the

[5] https://en.wikipedia.org/wiki/Esri_grid

[6] http://cfconventions.org/

[7] https://www.unidata.ucar.edu/software/netcdf/docs/building_netcdf_fortran.html

legacy output formats. However, this remains on the development roadmap, and may be a required speed optimization for the analysis of stochastic climate data in the DAFNE framework.

4. CURRENT STATE OF IMPLEMENTATION PROGRESS

In this section we outline the current state of implementation progress at the time of writing this report. The model is under active development so this description is a "moving target". Nevertheless, we introduce the already implemented features, and next steps based on their priority for inclusion in the DAFNE integrated WEF model.

4.1 IMPLEMENTED FEATURES

The features of the model that have already been implemented are discussed in some detail in sections 3.2 and 3.3. Here we simply provide a summary list of the main components implemented in the model core for ease of reference, and to put the contents of section 4.2 into context:

- Implemented Fortran class hierarchy of cells and specialized cells (overland, soil, channel) used to compose catchment models;
- Implemented machinery to create, manage and compose cells into stacks (or layers) and catchments, as well as catchments into system models;
- Implemented analytic solutions of (specialized) ODEs, and Muskingum-Cunge-Todini channel routing to solve water fluxes between cells;
- Implemented evapotranspiration algorithms;
- Implementation of engineering control elements (diversions, water abstraction, irrigation, lake/reservoir operation);
- Implemented methods to parse and ingest model setup and input files, and use this information to build the model;
- Implemented methods to ingest forcing data from file (including interpolation from stations when required);
- Implemented overall flow control to run model simulations (main loop);
- Implemented logging of simulation progress and diagnostic information, as well as storing simulation results to disk.

4.2 PLANNED FEATURES IN THE NEXT PHASE

The following features are due to be included in the new model code during the coming months in order to complete the most substantial part of code development for the integrated WEF model (D3.5). These have been split into highest priority tasks (must be implemented to meet the vision of the DAFNE WEF model), and low(er) priority tasks (still of significant importance, but less critical in the short term).

High priority

- Complete porting the sediment transport modules tested in the development version of TOPKAPI-ETH to the new model core (hillslope erosion, bedload transport and suspended sediment transport, as described in section 2.3);
- Implement the conservative solute transport concepts from WATET (section 2.2.1) into the new model core;
- Develop parallel execution capabilities to take advantage of modern multi-core processors, and high-performance computing clusters.

Low(er) priority

- Improve automated algorithm tests for the MCT solver, radiation and evapotranspiration process;

- Implement tests for control elements (diversions, water abstraction, irrigation, lake/reservoirs);
- Extend/improve API to model core, for linking to external models (e.g. AquaCrop, General Lake Model);
- Implement non-conservative solute transport concepts based on C-Q relationships (see section 2.2.2);
- Port snow-melt and glacier processes from TOPKAPI-ETH to new model core[8].

5. CONCLUSIONS

In this report we presented the conceptualisation of the modifications necessary to account for transport processes and sediment dynamics in spatial and temporal explicit fashion. We also introduced the extensive work that is in progress to re-engineer for this purpose an existing physically-explicit hydrological model (TOPKAPI-ETH), with the goal of meeting the significant computational modelling demands imposed by the WEF nexus modelling framework, which is one of the key components of the DAFNE project implementation.

The extended and redesigned hydrological model will form the flexible core component of the integrated WEF nexus model of deliverable D3.5 due for delivery in month 36. The programming developments have been advanced in parallel to advancing research and development activities to include new processes in the model for the spatially distributed simulation of solute and sediment transport at the scale of large catchments. These processes are formulated in the model both spatially and temporally variable because they are strongly controlled by agricultural expansion and operational practices, as well as the construction and operation of reservoirs for hydropower and water supply, and are therefore important to consider in an analysis of the WEF nexus.

In accordance with the deliverable description from the DoA, this report focussed on the description of the conceptualisation of the modified distributed hydrological model accounting for transport processes and sediment dynamics in a spatially and temporally explicit fashion.

This report gave a description of the modified model along with the current and planned implementation of such components. The introductory section provided some context for the work. Then, in the remainder of the report we outlined the hydrological transport component that will be used to simulate the water quality dynamics at the river basin scale with respect to solutes, either mobilised by water flows or by sediment transport. These are core active research and development activities under WP3 (section 2). The implementation of these transport related water quality components into the new TOPKAPI-ETH version is one of the goals of the activities around this deliverable (D3.1) and the upcoming D3.5. We also described the software development work underway to redesign the TOPKAPI-ETH hydrological model, making it more computationally efficient, flexible and portable (section 3). In section 4, we detailed the progress to date, and our next steps as we build the hydrological model and work to integrate and couple the other modelling components of DAFNE (e.g. Lake process modelling MS22; Agricultural modelling MS27).

The work reported here (and ongoing) is planned as a means of tying together the various models, which rely on the system water balance dynamics to compute indicators derived from the value of the various competing uses of water within the WEF nexus for the DAFNE case-study catchments.

[8] This is not strictly required for the applications to the DAFNE case studies.

6. REFERENCES

Albrecht T.R., Crootof A. and C.A. Scott, (2018) The Water-Energy-Food Nexus: A systematic review of methods for nexus assessment, *Environmental Research Letters*, doi:10.1088/1748-9326/aaa9c6

Aksoy H. and M.L. Kavvas, (2005) A review of hillslope and watershed scale erosion and sediment transport models, *Catena*, 64(2), 247-271

Arnold J.G., Allen P.M., and G. Bernhardt, (1993) A comprehensive Surface-Groundwater Flow Model, *Journal of Hydrology*, 142(1993), 47-69

Avery S., (2012) *Lake Turkana and the lower Omo: hydrological impacts of major dam and irrigation development*, African Studies Centre, University of Oxford

Basu N.B., Destouni G., Jawitz J.W., Thompson S.E., Loukinova N.V., Darracq A., Zanardo S., Yaeger M., Sivaplana M., Rinaldo A., and P.S.C. Rao, (2010) Nutrient loads exported from managed catchments reveal emerging biogeochemical stationarity, *Geophysical Research Letters*, 37, L23404, doi:10.1029/2010GL045168

Beasley D.B., Huggins L.F. and A. Monke, (1980) ANSWERS: A model for watershed planning, *Transactions of the ASAE*, 23(4), 938-944

Benettin P., van der Velde Y., van der Zee S.E.A.T.M., Rinaldo A., and G. Botter, (2013) Chloride circulation in a lowland catchment and the formulation of transport by travel time distribution, *Water Resources Research*, 49, 4619–4632, doi:10.1002/wrcr.20309

Benettin P., Bailey S.W., Campbell J.L., Green M.B., Rinaldo A., Likens G.E., McGuire K.J. and G. Botter, (2015) Linking water age and solute dynamics in streamflow at the Hubbard Brook Experimental forest, NH, USA, *Water Resources Research*, 51:9256-9272

Benettin P., Bailey S.W., Rinaldo A., Likens G.E., McGuire K.L. and G. Botter, (2017) Young runoff fractions control streamwater age and solute concentration dynamics, *Hydrological Processes*, 31, 2982-2986

Betrie G.D., Mohamed Y.A., van Griensven A. and R. Srinivasan, (2011) Sediment management modelling in the Blue Nile Basin using SWAT model, *Hydrology and Earth System Sciences*, 15(3), p.807

Beven K., (2012) *Rainfall-Runoff Modelling*, 2nd ed., 457 pp., Wiley-Blackwell, Chichester, U. K.

Bicknell B.R., Imhoff J.C., Kittle J.L., Donigian A.S. and R.C. Johanson, (1993) *Hydrological Simulation Program – Fortran (HSPF): Users Manual for Release 10*, U.S. EPA Environmental Research Laboratory, Athens, Georgia.

Birkel C. and C. Soulsby, (2015) Advancing tracer-aided rainfall-runoff modelling: a review of progress, problems and unrealized potential, *Hydrological Processes*, doi:10.1002/hyp.10594

Botter G., Settin T., Marani M. and A. Rinaldo, (2006) A stochastic model of nitrate transport and cycling at basin scale, *Water Resources Research*, 42(4)

Botter G., Bertuzzo E. and A. Rinaldo, (2010) Transport in the hydrologic response: Travel time distributions, soil moisture dynamics, and the old water paradox, *Water Resources Research*, 46, W03514, doi:10.1029/2009WR008371

Botter G., Bertuzzo E. and A. Rinaldo, (2011) Catchment residence and travel time distributions: The master equation, *Geophysical Research Letters*, 38, L11403, doi:10.1029/2011GL047666

Botter M., Burlando P. and S. Fatichi, (in review). Anthropogenic and catchment characteristics signatures in the water quality of Swiss rivers: a quantitative assessment, *Hydrology and Earth System Science*

Brutsaert W., (2005) *Hydrology. An Introduction*, Cambridge University Press, Cambridge, UK.

Carenzo M., Pellicciotti F., Rimkus S., and P. Burlando, (2009) Assessing the transferability and robustness of an enhanced temperature-index glacier melt model, *J. Glaciol.*, 55 (190), 258–274

Ciarapica L. and E. Todini, (2002) TOPKAPI: a model for the representation of the rainfall-runoff process at different scales, *Hydrological Processes*, 16(2), 207-229

Ding X., Shen Z., Hong Q., Yang Z., Wu X., and R. Liu, (2010) Development and test of the export coefficient model in the upper reach of the Yagtze River, *Journal of Hydrology*, 383, 233-244

Duncan J.M., Band L.E. and P.M. Groffman, (2017) Variable nitrate Concentration-Discharge Relationships in a Forested Watershed, *Hydrological Processes*, 31:1817-1824, doi:10.1002/hyp.11136

Elliot W.J. and J.M. Laflen, (1993) A process-based rill erosion model, *Transactions of the American Society of Agricultural Engineers*, 36 (1), 35–72

Fatichi S., Rimkus S., Burlando P., Bordoy R. and P. Molnar, (2015) High-resolution distributed analysis of climate and anthropogenic changes on the hydrology of an Alpine catchment, *Journal of Hydrology*, doi:10.1016/j.jhydrol.2015.03.036

Foster G.R. and L.D. Meyer, (1972) Transport of soil particles by shallow flow, *Transactions of the ASAE*, 15(1), pp.99-0102

Francipane A., Ivanov V.Y., Noto L.V., Istanbulluoglu E., Arnone E. and R.L. Bras, (2012) tRIBS-Erosion: a parsimonious physically-based model for studying catchment hydro-geomorphic response, *Catena*, 92, 216–231, doi:10.1016/ j.catena.2011.10.005

Gall H.E., Park J., Harman C.J., Jawitz J.W., and P.S.C. Rao, (2012) Landscape filtering of hydrologic and biogeochemical responses in managed catchments, *Landscape Ecology*, 28, 651–664

Godsey S.E., Kirchner J.W. and D.W. Clow, (2009) Concentration-discharge relationships reflect chemostatic characteristics of US catchments, *Hydrological Processes*, 23 (13), pp. 1844-1864

Godsey S.E., Aas W., Clair T.A., de Wit H.A., Fernandez I.J., Hahl J.S., Malcolm I.A., Neal C., Neal M., Nelson S.J., Norton S.A., Palucis M.C., Skjelvale B.L., Soulsby C., Tetzlaff D., and J.W. Kirchner, (2010) Generality of fractal 1/f scaling in catchment tracer time series, and its implications for catchment travel time distributions, *Hydrological Processes*, 24, 1660–1671, doi:10.1002/hyp.7677

Green, W. and G. Ampt, (1911), Studies on soil physics, part 1, the flow of air and water through soils, *J. Agric. Sci.*, 4, 1-24

Hanley N., Faichney R., Munro A. and J.S. Shortle, (1998) Economic and environmental modelling for pollution control in an estuary, *Journal of Environmental management*, 52(3), pp.211-225

Harman C.J., (2015) Time-variable transit time distributions and transport: Theory and application to storage-dependent transport of chloride in a watershed, *Water Resources Research*, 51, 1–30, doi:10.1002/2014WR015707

Heimann F.U.M., Rickenmann D., Böckli M., Badoux A., Turowski J.M. and J.W. Kirchner, (2015), Calculation of bed load transport in Swiss mountain rivers using the model sedFlow: proof of concept, *Earth Surface Dynamics*, 3(1), p.35

Hinderer M., Kastowski M., Kamelger A., Bartolini C. and F. Schlunegger, (2013), River loads and modern denudation of the Alps – a review, *Earth-science reviews*, 118, pp.11-44

Holvitie J., Licorish S.A., Spínola R.O., Hyrynsalmi S., MacDonell S.G., Mendes T.S., Buchan J. and V. Leppänen, (2018) Technical debt and agile software development practices and processes: An industry practitioner survey, *Information and Software Technology*, 96, 141-160

Hrachowitz M., Benettin P., van Breukelen B.M., Fovet O., Howden N.J.K., Ruiz L., van der Velde Y., and A.J. Wade, (2016) Transit times – The link between hydrology and water quality at the catchment scale, *WIREs Water*, doi:10.1002/wat2.1155

Hrachowitz M., Savenjie H., Bogaard T.A., Tetzlaff D., and C. Soulsby, (2013) What can flux tracking teach us about water age distribution patterns and their temporal dynamic? *Hydrology and Earth Systems Sciences*, 17(2), 533–564, doi:10.5194/hess-17-533-2013

Hutton C., Wagener T., Freer J., Han D., Duffy C. and B. Arheimer, (2016) Most computational hydrology is not reproducible, so is it really science? *Water Resources Research*, vol. 52, 7548-7555, doi:10.1002/2016WR019285

Kilinc M., (1972) *Mechanics of soil erosion from overland flow generated by simulated rainfall*, Phd thesis, Colorado State University.

Kirchner J.W., Feng X. and C. Neal, (2001) Catchment-scale advection and dispersion as a mechanism for fractal scaling in stream tracer concentrations, *Journal of Hydrology*, 254, 81–100

Kirchner J.W., (2006) Getting the right answers for the right reasons: Linking measurements, analyses, and models to advance the science of hydrology, *Water Resources Research*, 42, W03S04, doi:10.1029/2005WR004362

Kirchner J.W., Tetzlaff D. and C. Soulsby, (2010) Comparing chloride and water isotopes as hydrological tracers in two Scottish catchments, *Hydrological Processes*, 24(12), 1631–1645

Knisel W.G., (1980) *CREAMS: a field scale model for Chemicals, Runoff, and Erosion from Agricultural Management Systems*, United States. Dept. of Agriculture. Conservation research report (USA).

Konz M., Chiari M., Rimkus S., Turowski J.M., Molnar P., Rickenmann D. and P. Burlando, (2011) Sediment transport modelling in a distributed physically based hydrological catchment model, *Hydrol. Earth Syst. Sci.*, doi:10.5194/hess-15-2821-2011

Krysanova V. and Arnold J.G., (2008), Advances in ecohydrological modelling with SWAT – a review, *Hydrological Sciences Journal*, 53(5), pp.939-947

Lautze J., Phiri Z., Smakhtin V. and D. Saruchera, (2017) *The Zambezi River Basin: water and sustainable development*, Routledge

Liu Z., and E. Todini, (2005) Assessing the TOPKAPI non-linear reservoir cascade approximation by means of a characteristic lines solution, *Hydrol. Processes*, 19 (10), 1983–2006, doi:10.1002/hyp.5662

McDonnell J.J., McGuire K., Aggarwal P., Beven K.J., Biondi D., Destouni G., Dunn S., James A., Kirchner J.W., Kraft P., Lyon S., Maloszewski P., Newman B., Pfister L., Rinaldo A., Rodhe A., Sayama T., Seibert J., Solomon K., Soulsby C., Stewart M., Tetzlaff D., Tobin C., Troch P., Weiler M., Western A., Wörman A., and S. Wrede, (2010) How old is streamwater? Open questions in catchment transit time conceptualization, modelling and analysis, *Hydrological Processes*, 24, 1745–1754, doi:10.1002/hyp.7796

McDonnell J.J., (2017) Beyond the water balance, *Nature Geoscience*

McGuire, K. and McDonnell, J. J. (2006). A review and evaluation of catchment transit time modelling. Journal of Hydrology, 330 (3–4), 543–563.

McGuire, K., McDonnell, J. J., Weiler, M., Kendall, C., McGlynn, B. L., Welker, J. M., and Seibert, J. (2005). The role of topography on catchment-scale water residence time. Water Resources Research, 41, W05002, doi:10.1029/2004WR003657.

Merritt W.S., Letcher,R.A. and A.J. Jakeman, (2003) A review of erosion and sediment transport models, *Environmental Modelling & Software*, 18(8), pp.761-799

Meyer-Peter E. and R. Müller, (1948) *Formulas for bed-load transport*, In IAHSR 2nd meeting, Stockholm, appendix 2. IAHR.

Mitas L. and Mitasova, H. Mitasova, (1998) Distributed soil erosion simulation for effective erosion prevention, *Water Resources Research*, 34(3), pp.505-516

Moatar F., Abbott B.W., Minaudo C., Curie F., and G. Pinay, (2017) Elemental properties, hydrology, and biology interact to shape concentration-discharge curves for carbon, nutrients, sediment, and major ions, *Water Resources Research*, 53, 1270–1287, doi:10.1002/2016WR019635

Molnar P., Burlando P., Kirsch J. and E. Hinz, (2006) Model investigations of the effects of land-use changes and forest damage on erosion in mountainous environments, *IAHS-AISH publication*, pp.589-600

Moquet J.S., Guyot J.L., Crave A., Viers J., Filizola N., Martinez J.M., Oliveira T.C., Sanchez L.S.H., Lagane C., Casimiro W.S.L., Noriega L., and R. Pombosa, (2016) Amazon River dissolved load: temporal dynamics and annual budget from the Andes to the ocean, *Environmental Science and Pollution Research*, 23, 11405-11429

Morgan R.P.C., Quinton J.N., Smith R.E., Govers G., Poesen J.W.A., Auerswald K., Chisci G., Torri D. and M.E. Styczen, (1998) The European Soil Erosion Model (EUROSEM): a dynamic approach for predicting sediment transport from fields and small catchments, *Earth surface processes and landforms*, 23(6), pp.527-544

Musolff A., Fleckenstein J.H., Rao P.S.C. and J.W. Jawitz, (2017) Emergent archetype patterns of coupled hydrologic and biogeochemical responses in catchments, *Geophysical Research Letters*, 44, 4143–4151

Nearing M.A., Foster G.R., Lane L.J. and S.C. Finkner, (1989) A process-based soil erosion model for USDA-Water Erosion Prediction Project technology, *Transactions of the ASAE*, 32(5), pp.1587-1593

Neitsch S.L., Arnold J.G., Kiniry J., and J.R. Williams, (2011) *Soil and water assessment tool theoretical docu-mentation (Version 2009)*, USDA Agricultural Research Service and Texas A&M Blackland Research Center, Temple, Texas

Pandey A., Chowdary V.M., Mal B.C. and M. Billib, (2008) Runoff and sediment yield modelling from a small agricultural watershed in India using the WEPP model, *Journal of Hydrology*, 348(3), pp.305-319

Pellicciotti F., Brock B., Strasser U., Burlando P., Funk M. and J. Corripio, (2005) An enhanced temperature-index glacier melt model including the shortwave radiation balance: development and testing for Haut Glacier d'Arolla, *Switzerland. J. Glaciol.*, 51 (175), 573–587

Pieri L., Bittelli M., Wu J.Q., Dun S., Flanagan D.C., Pisa P.R., Ventura F. and F. Salvatorelli, (2007) Using the Water Erosion Prediction Project (WEPP) model to simulate field-observed runoff and erosion in the Apennines mountain range, Italy, *Journal of Hydrology*, 336(1), pp.84-97

Priestley C., and R. Taylor, (1972) On the assessment of surface heat flux and evaporation using large-scale parameters, *Monthly Weather Rev.*, 100 (2), 81–92, doi:10.1175/15200493(1972)100<0081:OTAOSH>2.3.CO;2.

Prosser I.P. and P. Rustomji, (2000) Sediment transport capacity relations for overland flow, *Progress in Physical Geography*, 24(2), pp.179-193

Recking A., (2010) A comparison between flume and field bed load transport data and consequences for surface-based bed load transport prediction, *Water Resources Research*, 46(3)

Remondi F., Kirchner J.W., Burlando P. and S. Fatichi, (2018) Water flux tracking with a distributed hydrological model to quantify controls on the spatiotemporal variability of Transit Time Distributions, *Water Resources Research*, 5(4), 3081-3099

Renard K.G., Foster G.R., Weesies G.A. and J.P. Porter, (1991) RUSLE: Revised universal soil loss equation, *Journal of soil and Water Conservation*, 46(1), pp.30-33

Rickenmann D., (1990). BED LOAD TRANSPORT CAPACITY OF SLURRY FLOWS AT STEEP SLOPES. Phd dissertation, Swiss Federal Institute of Technology Zurich.

Rickenmann, D. (2001) Comparison of bed load transport in torrents and gravel bed streams, *Water Resources Research*, 37(12):3295

Rigon R., Bertoldi G. and T.M. Over, (2006), GEOtop: a distributed hydrological model with coupled water and energy budgets, *Journal of Hydrometeorology*, 7(3)

Rimkus S., (2013) *TOPKAPI-ETH documentation and user guide*, Technical report, ETH, Zurich.

Rinaldo A., Beven K.J., Bertuzzo E., Nicotina L., Davies J., Fiori A., Russo D. and G. Botter, (2011) Catchment travel time distributions and water flow in soils, *Water Resources Research*, 47, W07537, doi:10.1029/2011WR010478

Schröder A., (2000) WEPP, *EUROSEM, E-2D: results of applications at the plot scale*, In Soil Erosion (pp. 199-250). Springer Berlin Heidelberg

Sharpley A.N., (2006) *Modelling phosphorus movement from agriculture to surface water*, Pages 3-20 in Radcliff, D. E., Cabrera, M. L., Modelling phosphorus in the environment

Shen C. and M.S. Phanikumar, (2010) A process-based, distributed hydrologic model based on a large-scale method for surface-subsurface coupling, *Advances in Water Resources*, 33(12), 1524-1541

Soulsby C., Tetzlaff D. and M. Hrachowitz, (2009) Tracers and transit times: windows for reviewing catchment scale storage? *Hydrological Processes*, 23, 3503–3507, doi:10.1002/hyp.7501

Soulsby C., Birkel C., Geris J., Dick J., Tunaley C., and D. Tetzlaff, (2015) Stream water age distributions controlled by storage dynamics and nonlinear hydrologic connectivity: Modelling with high resolution isotope data, *Water Resources Research*, 51:7759–7776

Tegene B., (1992) *Erosion: its effects on properties and productivity of Eutric Nisosols in Gununo Area, Southern Ethiopia, and some techniques of its control*, pp. 173, African Studies Series, University of Bern, Institute of Geography

Tetzlaff D., Uhlenbrook S., Eppert S., and C. Soulsby, (2009) Does the incorporation of process conceptualization and tracer data improve the structure and performance of a simple rainfall-runoff model in a Scottish mesoscale catchment? *Hydrological Processes*, 22(14), 2461–2474.

Thompson S.E., Basu N.B., Lascurain J.J., Aubeneau A. and P.S.C. Rao, (2011) Relative dominance of hydrology versus biogeochemical factors on solute export across impact gradients, *Water Resources Research*, 47, W00J05, doi:10.1029/2010WR009605

van der Velde Y., de Rooij G.H., Rozemeijer J.C., van Geer F.C. and H.P. Broers, (2010) Nitrate response of a lowland catchment: On the relation between stream concentration and travel time distribution dynamics, *Water Resources Research*, 46(11)

van der Velde Y., Torfs P.J., van der Zee S.E.A.T.M. and R. Uijlenhoet, (2012) Quantifying catchment-scale mixing and its effect on time-varying travel time distributions, *Water Resources Research*, 48, W06536, doi:10.1029/2011WR011310

van Genuchten M.T.V., (1980) A closed-form equation for predicting the hydraulic conductivity of unsaturated soils, *Soil Science Society of America*, 44(5), 892-898

Von Gunten A., (1993) *Soil Erosion Processes in a Twin Catchment Set-up in Gununo Area*, Ethiopia Soil Conservation Research Project, Research Report 23.

Walton R.S. and H.M. Hunter, (1996) *Modelling water quality and nutrient fluxes in the Johnstone River Catchment, North Queensland*. In Hydrology and Water Resources Symposium 1996: Water and the Environment; Preprints of Papers (p. 425). Institution of Engineers, Australia.

Weigel G., (1986) *The soils of the Gununo Area*, Sidamo Research Unit, Ethiopia, Soil Conservation Research Project, Research Report 8

Wicks J.M. and J.C. Bathurst, (1996) SHESED: a physically based, distributed erosion and sediment yield component for the SHE hydrological modelling system, *Journal of Hydrology*, 175(1-4), pp.213-238

Wilcock P.R. and J.C. Crowe, (2003) Surface-based transport model for mixed-size sediment, *Journal of Hydraulic Engineering*, 129(2), pp.120-128

Wischmeier W.H. and D.D. Smith, (1978) Predicting rainfall erosion losses-a guide to conservation planning

WWAP (United Nations World Water Assessment Programme)/UN-Water (2018). The United Nations World Water Development Report 2018: Nature-Based Solutions for Water. Paris, UNESCO.

Wymore A.S., Brereton R.L., Ibarra D.E., Maher K. and W.H. McDowell, (2017) Critical zone structure controls concentration-discharge relationships and solute generation in forested tropical montane watersheds, *Water Resources Research*

Young R. A., Onstad C.A., Bosch D.D. and W.P. Anderson, (1989) AGNPS: a nonpoint-source pollution model for evaluating agricultural watershed, *Journal of Soil and Water Conservation*, 107, 1–16

7. APPENDIX A – TOPKAPI DEVELOPMENT REPORT

This appendix contains a lightly edited version of the initial review carried out on the TOPKAPI-ETH model, and is included here to improve the readability of the main report by avoiding too much technical detail in the latter. The contents of this appendix are referenced in sections 3.2, 3.3 and 4 as necessary.

7.1 RATIONALE

- Topkapi kinematics scheme with analytic resolution of non-linear reservoir equations;

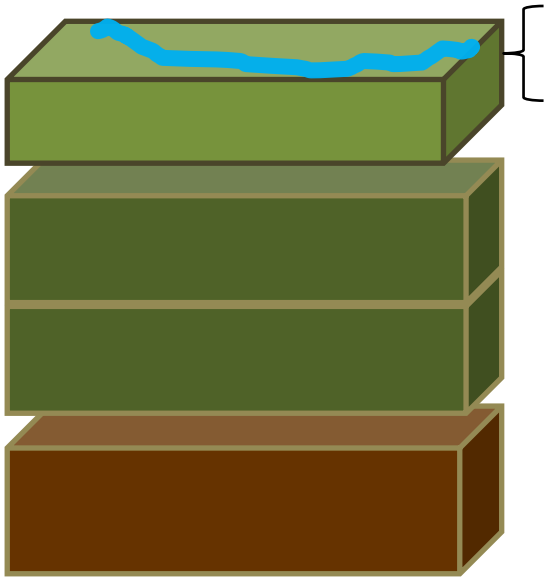

Figure 14 Configuration of the various stores in a TOPKAPI-ETH model cell.

- Resolution: 4D scheme at basin scale (for each cell: 0-3 in / 0-1 out in NS, SN, WE or EW direction);
- Useful in “nested-model-chain” framework to simulate main spatial-distributed physical processes for water planning and management purpose

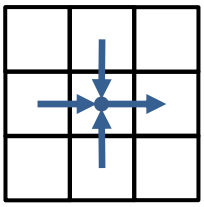

Figure 15 Illustration of the TOPKAPI ETH D4 drainage scheme.

7.2 CURRENT VERSION ANALYSIS

History of versions:

- PROGEA original code
- v32 (EU project ACQWA)
- v64 (new organization of the code)

Todini's origins must be acknowledged but the research code also needs to fork out from the Topkapi community and the Mazzetti/Progea commercial version [http://www.progea.net/prodotti.php?p=TOPKAPI].

Any contact with the original developer (Stefan Rimkus) is possible via Simone Fatichi. The latest official release is August 2013.

Documentation is partly constituted by excel tables that specify the I/O formats and the meaning of the major variables. However, this material is to be checked even if it should cover at least 90% of the code.

There is also a so-called “light version”: probably compiled for Paolo Burlando (by Cinzia Mazzetti) for academic purposes.

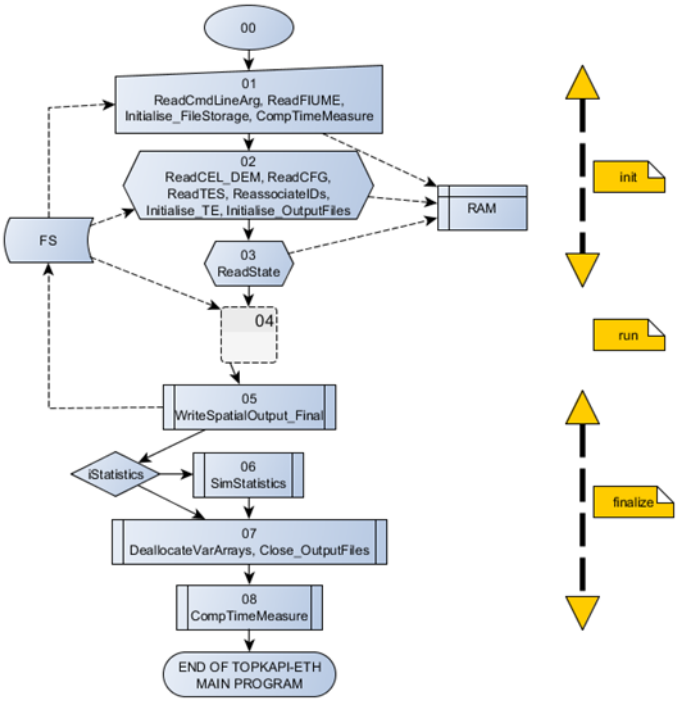

Figure 16 Main processes when TOPKAPI-ETH is run.

Reconnaissance (see file “topkapi.xslx”):

TOPKAPI SOURCES

Path	Description	Status	Reference	Input(s)	Output(s)	Called Fun	ToDo	File	Comment
root	global variables + main								
Artificial_Structures									
Bedload_Transport									
Channel									
		unused						CalcStricklerRoughness_HWRM.for	Compute Strickler coefficients
		unused	Todini, E. [2007]			scalaX, qdy		CHANNEL_Musk_Cun_Tod.for	It solves the channel compon method for a channel with a g
		to be checked	Mazzetti C	param =	q_output =	An_sol_ET		CHANNEL_Res_Rect.for	Solves the channel componer RECTANGULAR section
		to be checked	Mazzetti C. and Todini E. [200?]			An_sol_ET		CHANNEL_Res_Tri.for	Solves the channel componer section
		ok			hdv: Depth	NONE		hdv.for	Given a generic section and a the lowest point of the riverb
		ok						qdy.for	Given a generic section and a section width at water surfac
		unused		param : Ch	y : Depth of water (from the lov			ScalaX.for	Given a generic section and a (Bx), wet area (Ax) and celerit

Figure 17 Screen shot of the spreadsheet containing the analysis of the TOPKAPI-ETH algorithms and source files.

- list of steps in main program (see Figure 17 and xls sheet “main”);
- parameters array definition (see xls sheet “param”);
- list of subdirectory and files with status and comments (see xls sheet “files”);
- list of all global variables and their use (see xls sheet “variables”);
- i/o interfaces (see xls sheet “IO”);

- analysis of variables initializations (see xls sheet “initialize”);
- v32 vs v64 files comparison (see xls sheet “old_vs_new”);
- profiling (see forward).

7.2.1 Main drawbacks

- mixed programming and comment styles caused by different authors in different years;
- mixed logic flow in the program: some parts in main program, some parts in called procedures;
- use of a huge number of global variables without write protection;
- some constant is locally redefined (see for example “pim” in file `Channel/ScalaX.for#78`);
- no R/W separation between procedure arguments;
- no handling exceptions in I/O and run-time operation;
- no real modules separation (only subfolders organization in 64 bit version);
- hide and distributed structural hypothesis (e.g. D4 connection between cells; possible “types” of cells such as lake or glacier; hard coded “`fiume`” file to start configuration phase; …);
- some procedures are redefined (such as “Saturation vapour pressure” in `Radiation/CalcGI_HWRM.for#486` and `ET/CalcETP_HWRM.for#148`);
- some test (“if” instructions) uses “=” between reals without tolerance; some others use “if/elseif” statement without “else”;
- use of real variables to switch between subcases in “select case” statement; in some “select case”, “default” is missed;
- use of non-initialized variables (see for instance “`sCWH`” in `Snow_Glaciers/CalcSnowIceMelt_HWRM.for#400`, or “`y`” in `Channel/ScalaX.for#123`);
- use of needless variables (see “`PsiWF`” in `Surface/CalcInfiltration_HWRM.for#104`, for instance);
- command line arguments are present but not used;
- ambiguous transformation from nominal surface (horizontal) and real surface using S2Area-Ratio parameter (for instance commented in `ET/CalcETA_HWRM.for#96`);
- not all variables and parameters unit of measures are specified;
- some “modules” use unclear constructs (see for instance `Snow_Glaciers`)
- use of “9999” and similar as “no info” code.

7.2.2 Profiling

Heap profiling, time profiling, graph profiling, call profiling (see file “profile.xlsx”):

1. “**timeloop_outputctrl**” in `Save_Data/TimeLoop_OutputCTRL_HWRM.for` spends about 12% of the total time to save results on files;

- Called by main in 04.17. (AGGREGATE STATE OF CATCHMENTS AND COMPUTE WATER BALANCE [SR])

2. “**an_sol_et_**” in `Scientific_Functions/An_sol_ET.for` spends about 8.5%-11%, called more than 7 billion of time during a 1 year length simulation!

- Called in internal cycle over time and space from:
  - soil_res_et_ (`SubSurface/SOIL_Res_ET.for`) – in 04.10.05. (SOIL MODULE - LOWER SOIL LAYER, SOILB) e 04.10.07. (SOIL MODULE - TOP SOIL LAYER, SOILA)
  - channel_res_rect_ (`Channel/CHANNEL_Res_Rect.for)` - 04.10.08.02.02.02. (CHANNEL MODULE - CASE: NON-LINEAR RESERVOIR)
  - surf_res_et_ (`Surface/SURF_Res_ET.for`) - 04.10.08.02.01. (SURFACE MODULE)

3. “**calcsnowicemelt_**” in `Snow_Glaciers/CalcSnowIceMelt_HWRM.for` spends about 9%. Called from the main (in the time-cycle); the cycle over the space is done in the function.
4. “**populateparamcurrentgc_**” in `PopulateParamCurrentGC_HWRM.for` spends about 4-5%. Called from the main for each time-step of the simulation.

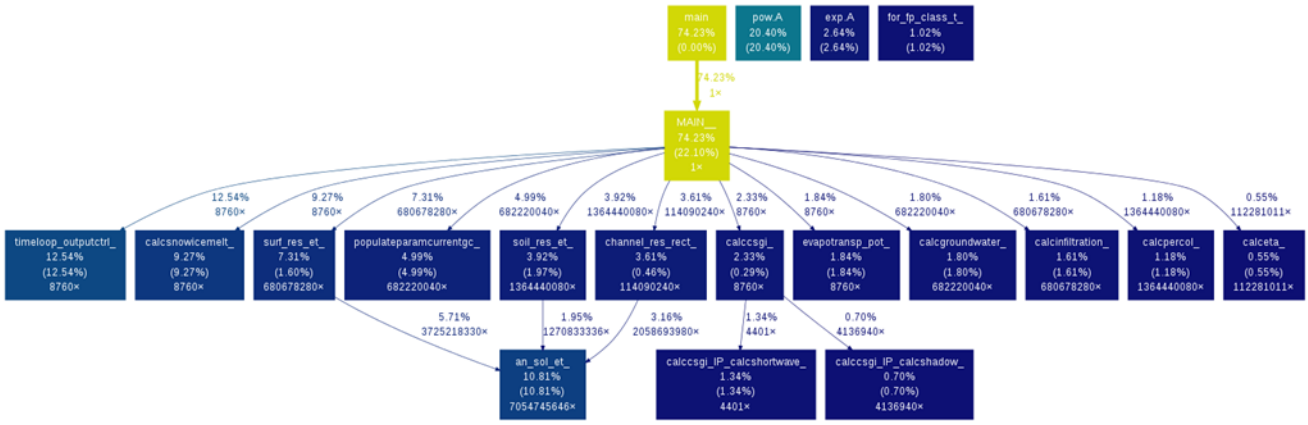

Figure 18 Typical profiling results for the existing version of TOPKAPI-ETH on a 64-bit Windows computer.

7.2.3 Analysis of some specific functionalities and minor extensions

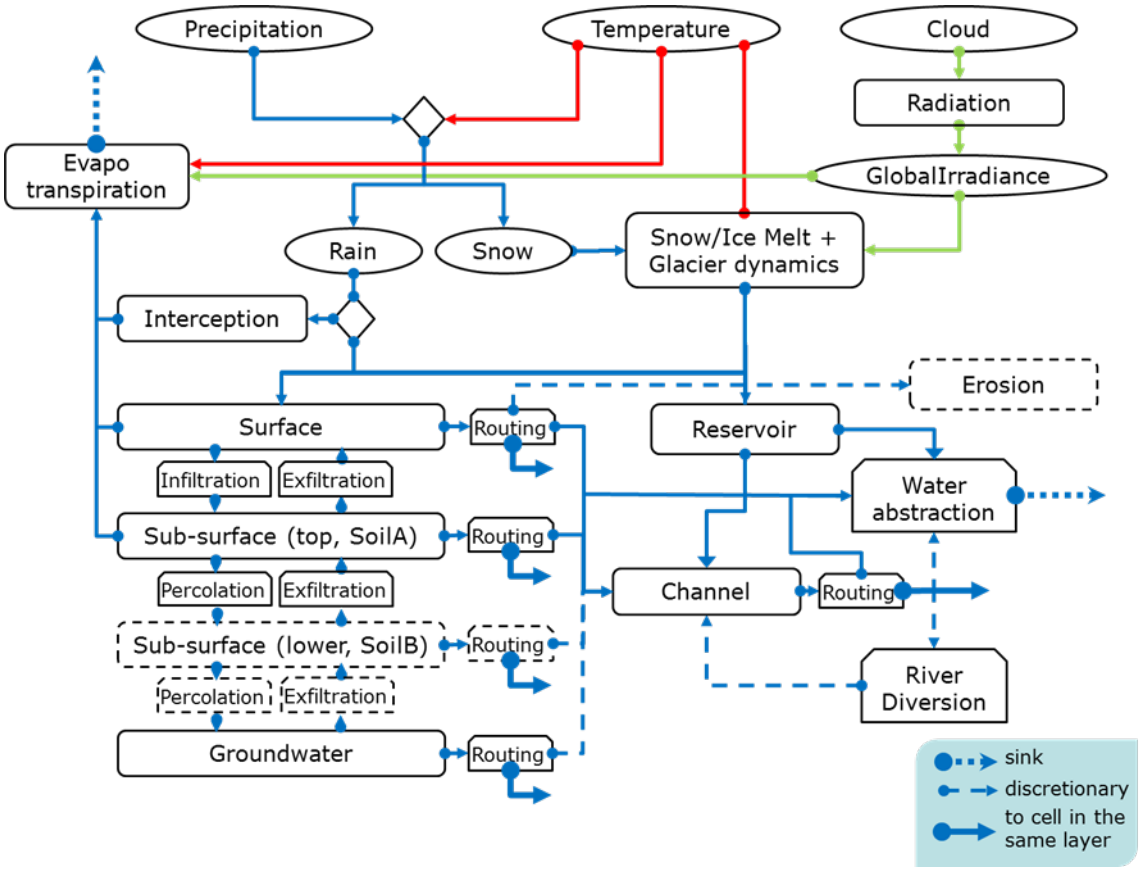

Figure 19 Processes modelled and their inter-linkages in TOPKAPI-ETH.

Orography

3 different variables:

- DTM_K2: vector of elevation without ice thickness [m a.s.l], read from "*.tes" file (see `Load_Data/ReadTES_HWRM.for#120`);
- Elev_2D: matrix of elevation of each cell [m a.s.l.], read from "*.dem" file (see `Load_Data/ReadCEL_DEM_HWRM.for#154`);

- Elev_K2: vector of elevation including ice thickness [m a.s.l.], obtained from DTM_K2 (see `Load_Data/ReadTES_HWRM.for#274`);

Matrix is used to cover peripheral area of the simulation domain to obtain cells terrain shading.

Precipitation and Air Temperature Interpolation

Interpolation of precipitation code in `Scientific_Functions/CalcPrecEWE_HWRM.for`:

- seasonal approach
  - `Prec_K2 = Prec_K2 + Prec_K2*PEMF1_2D` (month 4-9)
  - `Prec_K2 = Prec_K2 + Prec_K2*PEMF2_2D` (other months)
- global approach
  - `Prec_K2 = Prec_K2*PEMF1_2D`
- monthly correction
  - `Prec_K2 = Prec_K2*PEwe_K2_MM(mm)`

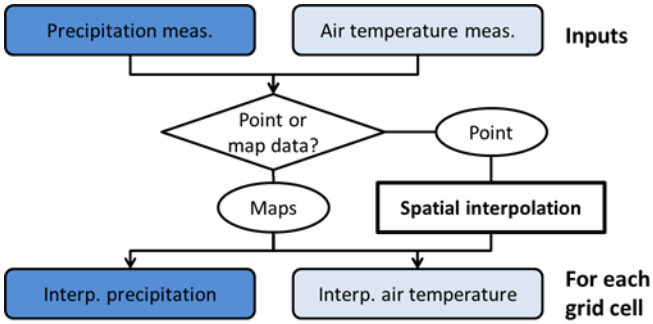

Figure 20 Flow chart indicating the decision points for interpolation of precipitation and air temperature measurements ingested by TOPKAPI-ETH.

First interpolation method uses hard coded months 4 and 9 to separate seasonal intervals;

Interpolation of temperature code in `Load_Data/GetTemp.for`:

- using time invariant temperature lapse rate
  - `Temp_K2 = xTemp1D(kTGauge)+Tgrad*(ZTGauge(kTGauge) - Elev_K2)`
- using monthly temperature lapse rates per gauge
  - `Temp_K2 = xTemp1D(kTGauge)+xTgrad(kTGauge,mm)*(ZTGauge(kTGauge)-Elev_K2)`
- using monthly temperature correction factors maps
  - `Temp_K2 = xTemp1D(kTGauge)*TGradMonthly_K2(mm)`
- using air temperature lapse rate time series
  - `Temp_K2 = xTemp1D(kTGauge)+xTgrad1D(kTGauge)*(ZTGauge(kTGauge)-Elev_K2)`

and `TOPKAPI_HWRM.for` (in cases 2, 3 and 4):

- to modulate air temperature over glaciated areas:
  - `Temp_K2 = max(Temp_K2 - Tmod_OnGlacier, ZeroValue)`
- to modulate air temperature over debris covered parts of the glacier:
  - `Temp_K2 = Temp_K2 + Tmod_OnDebris`

Evapotranspiration

Potential ET, code in `ET/CalcETP_HWRM.for`:

- Modified Progea approach
- Makkink I
- Makkink II
- Priestly-Taylor

Makkink formulations are to be checked (or probably removed)

All methods are corrected considering a crop factor, depending from a time-invariant land use:

`ETp_K2 = cropf(Luse_K2)*ETPnocf_K2`

Actual ET, code in `ET/CalcETA_HWRM.for`:

Comparison with water available from rain: take water from surface volume and then from soil volume

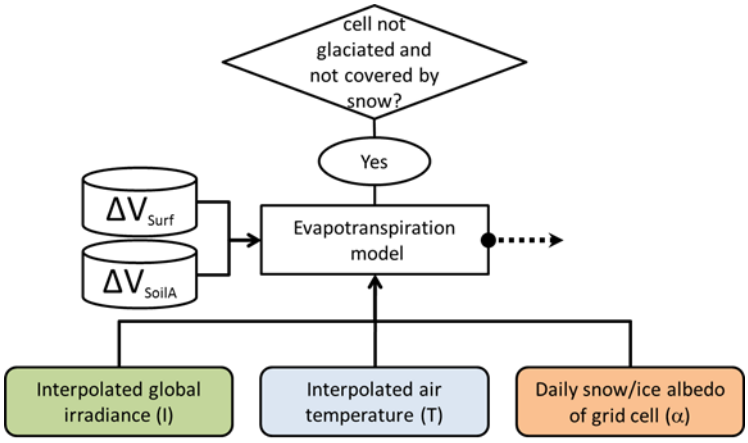

Figure 21 Flow chart indicating the decision points and inputs required for the evapotranspiration module of TOPKAPI-ETH.

Interception

Simple bucket model, with max interception volume, obtained considering leaf area index (LAI) and max water height on leaf surface (code in `Surface/CalcInterception_HWRM.for`). All parameters are 'nominal', constant over the whole horizon.

Infiltration

2 different kind of models:

- Infiltration excess approach, code in `TOPKAPI_HWRM.for`
- Green-Ampt infiltration approach, code in `Surface/CalcInfiltration_HWRM.for`

Second method is an "event-driven" simulation, as melt process one.

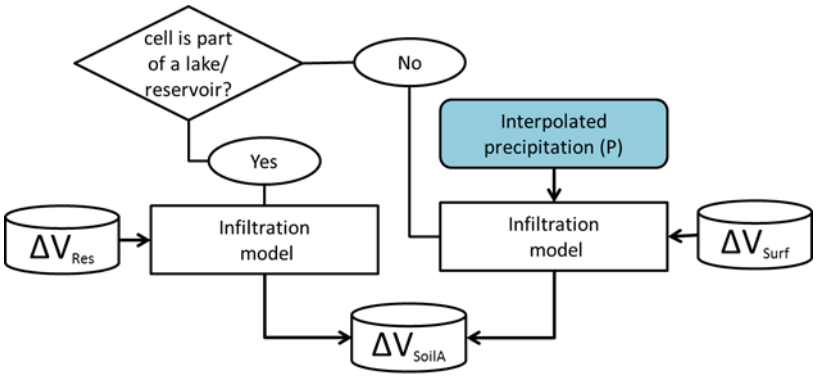

Figure 22 Flow chart indicating the decision points and inputs required for the infiltration module of TOPKAPI-ETH.

Non-linear reservoirs analytic solution for Channel, Surface and Sub-surface components

Equation:

dy/dt = a - b*y**c -> dy/dt = AA*(y**2 + BB*y + CC)

Code in:

- `Scientific_Functions/An_sol_ET.for`
  - SOLVE for a = 0. (q_in = 0)
  - SOLVE for a > 0. (q_in > 0): real and complex solution
  - SOLVE for a < 0. (q_in < 0): complex solution

Called from:

- `Channel/CHANNEL_Res_Rect.for`
  - `y` = channel level (`h_in`)
  - `AA = q_in/(wth_a**alpha/(wth_a + 2.d0*h_in)**alpham*xpix)`
  - `BB = slp**.5d0/(cmc_a*xpix)`
  - `CC = alpha = 5/3`
- `Channel/CHANNEL_Res_Tri.for`
  - `y` = channel volume (`v_in`)
  - `AA = q_in`
  - `BB = (slp**0.5d0)*(c**alpham*s**(2.d0/3.d0)/(cmc_a*2.d0**(2.d0/3.d0)*xpix**alpha))`
  - `CC = alpha = 4/3`
- `Surface/SURF_Res_ET.for`
  - `y` = initial water depth over the ground surface (`h_in`)
  - `AA = q_in/(wth*xpix)`
  - `BB = SigmaSurf*(slp**0.5d0)/(smcn*xpix)`
  - `CC = alpha`
- `SubSurface/SOIL_Res_ET.for`
  - `y = h_in = v_input/pixarea`
  - `AA = q_input/pixarea`
  - `BB = SigmaSoil*thick*xksh*slp/((dtheta**alpha)*(thick**alpha))/xpix`
  - `CC = alpha`

For channel and surface simulation (not for soil) is present an internal cycle over the time to integrate the analytic solution of the non-linear reservoir (code in `Channel/CHANNEL_Res_Rect.for#145` and `Surface/SURF_Res_ET.for#147`). The number of internal steps are fixed to 12 if celerity is equal to zero (without tolerance), or to "`dt*cel/xpix`", where "`dt`" is the main simulation step length and "`xpix`" is the cell dimension.

Only for channel, there is a further time-loop (`TOPKAPI_HWRM.for#1574`) using "`nchan`" steps. This variable is set to `dt00/dtch` (same file, `#272`), where `dt00=60*idt` and `dtch=60*idtch` (see `Initialize_Finalize/Initialise_TE_HWRM.for#126` and `#129`), and the latter are obtained from configuration file (`Load_Data/ReadCFG_HWRM.for#527` and `#529`.

Groundwater

The dynamic of this component is formalized with a linear reservoir and its analytic solution.

Equation:

```
Vt1 = Qin*(1 - exp(-k*dt00))/k + V2route*exp(-k*dt00)
```

Code in

- `SubSurface/CalcGroundwater_HWRM.for`

Routing simulation

Calculation of the surface and subsurface (top and lower soil layers) drainage coefficients to take in account cells partially covered by channel network (code in `Initialize_Finalize/Initialise_TE_HWRM.for`):

- `SigmaSurf_K2(ii) = 1. + sqrt(Slp_P_K2(ii)/Slp_K2(ii))`
  - `if (SigmaSurf_K2(ii).gt.5.) SigmaSurf_K2(ii)= 5.d0`
- No Sigma values provided in CFG file:
  - `if (SigmaSoil_TPK(kSoil_K2(ii)).lt.0.d0) then`
    - `SigmaSoil_K2(ii) = min(1. + Slp_P_K2(ii)/Slp_K2(ii),5.d0)`
- Sigma values provided in CFG file:
  - `else`

```
SigmaSoil_K2(ii) = SigmaSoil_TPK(kSoil_K2(ii))
```

Calculation of the width of the channel, code in

- `Load_Data/ReadTES_HWRM.for:`
  - triangular cross section:

```
RivTopWidth_K2 = 2.d0*RBWaterLevel_K2/tan(RivSideAng_K2)
```

  - rectangular cross section:

```
RivTopWidth_K2 = RivBedWidth_K2
```

  - trapezoidal cross section:

```
RivTopWidth_K2 = RivBedWidth_K2+2*RBWa-
terLevel_K2/tan(RivSideAng_K2)
```

- `TOPKAPI_HWRM.for`:
  - `wthsup = CellSize - RivTopWidth_K2 !< [m]`
- `PopulateParamCurrentGC_HWRM.for`:
  - `paramX(14) = RivBedWidth_K2(k2)*(CellSize - wthsup)`

Switch from non-linear reservoir simulation to Muskingum-Cunge-Todini method `if slope <= thresMC`.

In channel simulation, boundary values of wet surface, flow and volume ("`amax`", "`qmax`" and "`vmax`") are evaluated but not used. In the code (see comment in `Channel/qdy.for#20`), water level "`y`" is always referred to the bottom of the riverbed and, in case `y>ymax` (water in the floodplain), `q` is the discharge coming out of the floodplain, the total discharge should be `q+qmax`.

The function `Channel/CalcStricklerRoughness_HWRM.for` is unused: is present only in a commented line of the main program (`TOPKAPI_HWRM.for#1335`).

Lake/Reservoir simulation

Lake/Reservoir simulation is performed with code in `Lakes_Reservoirs/LakesReservoirs_HWRM.for`, but for inflow initialization, computed directly in `TOPKAPI_HWRM.for` (to take in account snow).

Main steps are:

1) Inflow evaluation, adding rain, channel and surface routing;
2) Evaporation, using a term obtained in main program (`TOPKAPI_HWRM.for#817`), as simple potential evaporation (`ETp_K2`) over all the cell area;
3) Max allowed volume constrain test and eventually overspill evaluation;
4) Environmental flow subtraction, as constant value obtained from the configuration file (`Load_Data/ReadCFG_HWRM.for#1377`);

5) Outflow evaluation, using different modes:
   a) Spillway discharge (distinguishing spillway section totally submerged or not), using function in file `Lakes_Reservoirs/CalcSpillwayDischarge_HWRM.for`;
   b) Interpolation and integration of a level/release table (code in `Lakes_Reservoirs/Res-Int_Tab.for`) read from the configuration file (see `Load_Data/Read-CFG_HWRM.for#1480`);
   c) Use of water level timeseries, read externally from specific file (`Load_Data/GetReser-voir.for`);
   d) Use of outflow timeseries, read externally from specific file (`Load_Data/GetReser-voir.for`);
   e) Use of a regulation policy (code in XXXX).
6) Volume balance

River diverting and pumping system simulation

River diverting system is simulated in the main code (`TOPKAPI_HWRM.for`), during channel routing evaluation, and considering 3 different subcases:
1) The cell is a take-out: the diverted flow can be calculated using 2 alternatives:
   a) as a ratio of the available flow in the cell, using a fixed value read from the configuration file (see `Load_Data/ReadCFG_HWRM.for#1610`);
   b) using a diversion policy to perform the simulation of a pumping system (see XXXX);
2) The cell is an intake;
3) The cell is a sink, with no return point and losing water, and flow obtained using a fixed ratio as in the subcase 1-a.

Water abstraction

Water abstraction is performed in 2 separated parts of the main code (`TOPKAPI_HWRM.for`): before and in the main time-loop:
- Before, to consider cell where surface is part of area with water abstraction, distinguish between:
  – Urban/Industry/Livestock, using analytic solution of a linear reservoir;
  – Agriculture, using moisture deficit in uppermost subsurface layer.
- During the time loop, to eventually reduce the volume in:
  – Channels;
  – Soil/subsurface layer A;
  – Soil/subsurface layer B;
  – Groundwater aquifer;
  – Lake/reservoir.

In both cases, the abstracted water flow is read externally from a timeseries file (see `Load_Data/GetWA_HWRM.for`)

Global Irradiance

Code in `Radiation/CalcGI_HWRM.for`

Time zone correction, day and hour angle, sun vector and solar zenith angle are obtained using only one average value of latitude, longitude, time-zone and sky visibility.

There is a comment (`Radiation/CalcGI_HWRM.for#173`) that must be understood: “Hour angle (omega) as in [Spencer, 1971]. Shift for summer time is not corrected. that has to be done in the input time series”

`CalcShadow`: calculation of Sun plain vector; Sun position relative to each cell is used to decide from which side of the area array the shade computation has to start: if x is negative, the sun is on the West and the calculation runs along grid columns; if y is negative, the sun is on the North and

>> beginning with grid rows. For each cell check if it is shaded by terrain (i.e. others cell) or not, when the cell is set as new shading origin.

`CalcShortwave`:

- Calc extra-terrestrial irradiance;
- Calc atmospheric properties and transmittance functions;
- Calc direct irradiance;
- Calc diffuse irradiance thought atmosphere after one and multiple passes;
- Finally calc global irradiance.
-

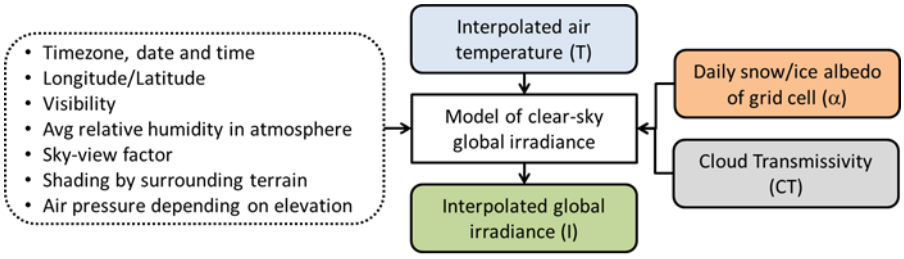

Figure 23 Flow chart indicating the decision points and inputs required for the global irradiance module of TOPKAPI-ETH.

Snow Albedo

2 different kind of parameterization (code in `Snow_Glaciers/CalcSnowIceMelt_HWRM.for`)

- Brock [2000]:
  - `Albedo_K2 = p1 - p2*log10(Tacc_K2)`
- Douville [1995]:
  - `Albedo_K2 = (Albedo_K2 - p2)*exp(-p3*dt00/86400.d0) + p2`
    (Melting condition)
  - `Albedo_K2 = Albedo_K2 - p4*dt00/86400.d0`
    (Frozen condition)

where "`Tacc_K2`" is the accumulated positive air temperature.

The albedo is obtained using an "event-driven" simulation: it is reset if precipitation rate is greater than albedo reset threshold or if the cell became snow covered during current timestep.

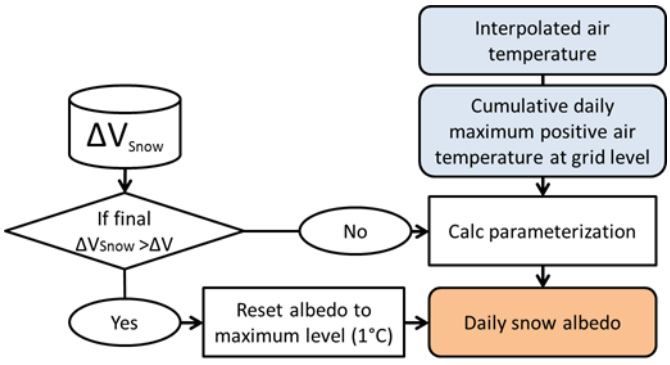

Figure 24 Flow chart indicating the decision points and inputs required for the snow albedo module of TOPKAPI-ETH.

Snow-pack process simulation

Developed by Silvan Ragettli and Francesca Pellicciotti (ex ETHZ).

The function is called by main program, each time-step, but outside the main cycle over space (cells). This process is an "event-driven" simulation too.

Steps of calculation (code in `Snow_Glaciers/CalcSnowIceMelt_HWRM.for`):

1) Evaluate the delay on the basis of cold content of the snow and ice pack per elevation band (as in figure);
2) Handle fluid/solid precipitation distinction and redistribution of solid precipitation along the main flow lines;
3) Gravitational redistribution of snow precipitation, using one of the following functions:
   a) `CalcSnowGravRedist()`, code in `Snow_Glaciers/CalcSnowGravRedist_HWRM.for`
   b) `CalcSnowSlide()`, code in `Snow_Glaciers/CalcSnowSlide_HWRM.for`
4) Snow & ice melt (loop over cells):
   a) Evaluate of catch redistributed snow and snow precipitation to route it later into the lake/reservoir;
   b) Retrieve variable & parameter values for albedo and melt computations;
   c) Reevaluate surface albedo;
   d) Compute snow and ice melt;
   e) Subcase: account for glacier thinning and eventual retreat;
   f) Subcase: 2nd snow module is enabled (account for snow melt retention)
      i) Melt retention in snow pack;
      ii) Refreezing in snow pack;
   g) Refreezing for balance;
   h) Glacier mass balance;

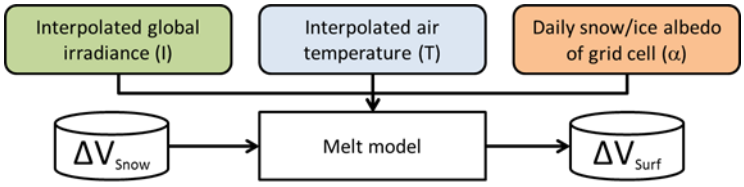

Figure 25 Flow chart indicating the decision points and inputs required for the snow pack process simulation module of TOPKAPI-ETH.

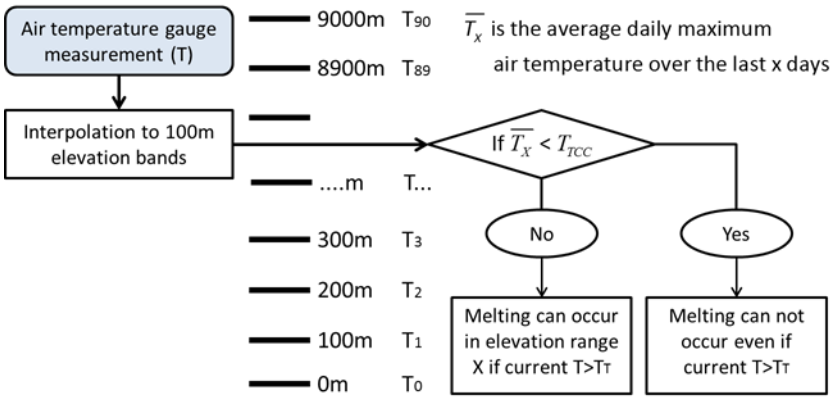

Figure 26 Flow chart indicating the decision points and inputs required for the snow melt model of the snow pack simulation module of TOPKAPI-ETH.

Some critical issues:
1) Melt process starts on the basis of an approximation of the temperature using a 100m band model (performed with an hard-coded loop over 90 bands - 90x100=9000 max possible elevations);
2) Band moving average temperature is obtained with a fixed dimension matrix (time x bands);
3) In the albedo reset subcase, the maximum daily temperature is obtained without check of the hour of the day (`Snow_Glaciers/CalcSnowIceMelt_HWRM.for`#257);
4) In the "glacier thinning and eventual retreat", a subcase of updating variable MeltIce_K2 produces a negative value (`Snow_Glaciers/CalcSnowIceMelt_HWRM.for#351`)
5) The Snow height of last year (`SttSnow_LY_K2`) is set 2 times as:
   `SttSnow_LY_K2 = min(SttSnow, SttSnow_LY_K2)`
   in `TOPKAPI_HWRM.for#2183` and `Snow_Glaciers/CalcSnowIceMelt_HWRM.for#461` but only if the cell is marked as glacier (`GlacID_K2>0`)
6) The "glacier thinning and eventual retreat" is controlled from switch "`iGlacThickness`", automatically set during TES file reading action. On the contrary from configuration file is possible to switch on or off the "Glacier Mass Balance" using switch "`iMapGMB`" but in the function the balance is always performed;

Bed load sediment erosion and transport simulation

Erosion simulation (code in `Erosion_Landslides/CalcErosion_HWRM.for`)

- Net erosion (if >0) or deposition (<0) rate [kg m-2 s-1] is obtained as:
- `DeE = (-qsedinE + qsedE)/CellSize`
- where "`qsedE`" can be obtained with 2 different simple approaches [KILINC, 1972]:
  - shear stress-based:
    - `tau = RhoWater*grav*H*Slope`
    - `if (tau.gt.taucritcell) qsedE = ktcell*((tau - taucritcell)**mucell)`
  - runoff-based:
- `qsedE = alphased(k2)/rohsed(k2)*(slope**betased)*(qspec**gammased)`

Transport simulation (code in `Bedload_Transport/CalcSedTransChPropagation_HWRM.for`)

- They are present 2 internal cycle over cells (so outside the space-loop of the main program) to perform sediment balance and to adjust slope.
- Balance can be obtained with 2 alternatives:
- Subgrid enabled: internal loop over cross sections, where relative data are read from a specific "*.cg2sedcs" file (see `Load_Data/ReadCFG_HWRM.for#2424`);
  - Subgrid disabled: no internal loops.
- In all cases balance is performed through following steps:
  1. Sediment routing from upper cells;
  2. Flow resistance due to form roughness (`nrntot`), specific critical discharge (`qcrit`) and sediment transport rates in the channel (`tcp`), using function CalcSedTransCh (code in `Bedload_Transport/CalcSedTransCh_HWRM.for`):
     - `nrntot` is obtained using 5 alternative methods:
       1. Rickenmann [1996], with torrents steeper 0.6%;
       2. Rickenmann [1996], with torrents steeper 0.8%;
       3. Rickenmann [2005a] slightly modified presented in Chiari [2010]
       4. Rickenmann [2005b];
       5. Rickenmann [2012].
     - Same number of different alternatives for `qcrit` evaluation, using sediment diameters read from the configuration file (see `Load_Data/ReadCFG_HWRM.for#2377` and following):

1. Schoklitsch [1950] using d40;
2. Bathurst [1985] using d50;
3. Rickenmann [1990] using d50;
4. Whittacker and Jaeggi [1986] using d65;
5. Rickenmann [1990] using d90.

- Finally, there are 3 methods for `tcp` evaluation:
  1. Rickenmann [1990];
  2. Rickenmann [2001];
  3. Recking [2010].

3. Set channel width of downstream cell;
4. Perform sediment balance considering sediment distribution along the channel length and the adjustment of sediment storage according to difference to upstream storage level.

- Slope adjustment too is performed with subgrid enabled or not, and following steps:
  1. downstream bed height;
  2. upstream bed height;
  3. calculation of x-length for slope determination;
  4. slopes updating.

7.3 NEW VERSION DESIGN AND FIRST IMPLEMENTATION

Target: a secure, fast, modularized and extensible code to run on a multi-core 'traditional' server (ex. 64 cores) with no RAM limitation.

Consolidation, increased usability and possibly increased execution speed of existing functions (including doctoral degrees and future sustainability) seems to be a priority as compared to the introduction of new functions, except for ice, solid transport and tracing.

Even from the point of view of parallelization it seems that the need to perform so many parallel processes in parallel (with incoming stochastic scenarios) is far ahead. It would therefore be worth thinking about an experimental configurator that allows you to track the inputs used with the results obtained (perhaps even the code you are using!) and allow you to launch it in parallel. In the future, it could be configured as a service that provides some results (such as solar radiation) to all parallel simulations, pre-calculating solutions once.

For DAFNE application it could be necessary to simulate 40 years (2010-50) for about 50 times.

7.3.1 How to enhance security

- No global variables;
- Explicit intent (only "*in*", "*out*" when possible, rarely "*inout*") of all procedure arguments;
- Use of "*NaN*" as "no data" code;
- Generalized handling exception;
- Specific data containers definition;
- Use of specific labels ("*enum*") to switch between subcases;
- Sharing only necessary functions and procedures among modules separating public part from private one;
- Array definition with *assumed-shape* hypothesis (when possible);
- @todo: use of special polymorphic containers (ex. PolyCon);
- @todo: systematic creation of specific – small – function, to implement automatic code test for each module functionalities.

7.3.2 How to enhance speed

- Separation between specialized calculous code and logic chain of flow program;
- Use of "*elemental"* (to support parallelization) and generic (to eventually downscale to single precision or upscale to quadruple precision) procedures;
- Introduce smart euler solver of non-linear dynamic models;
- Enhance adaptation of the time-step in internal process simulation;

- Asynchronous writing operation;
- Parallel simulation environment using master-slave paradigm;
- Array access: *smallest/fastest changing/innermost-loop index first*;
- Non-homogeneous spatial representation;
- Dynamic switch on/off some modules during simulation, freezing the state.

7.3.3 How to enhance flexibility

- Command line arguments support;
- Introduce specific "class" for:
  - **Cell**: space-variant neighboring function and dimension; space and time variant parameters; specific run function for modules and methods (*Decorator Pattern*)
  - **Layer**: constructor of cells and topology connections, created by main (*Factory Pattern*);
  - **Reader** (*Singleton Pattern*): to use different format for input files specified by templates or to eventually support remote reading;
  - **Writer** (*Singleton Pattern*): to support asynchronous writing, user-defined format by templates and eventually remote writing operation;
  - **Linker** (*Singleton Pattern*): to support logic provided by external services;
  - **Logger** (*Singleton Pattern*): to support error generation on different pipes and hierarchical messages;
- Time-variant parameterizations (for instance LAI parameter in the year, or land use along a multi-years simulation horizon, or minimum environmental flow from reservoirs);
- Space-variant parameterizations (for instance to adapt lat/long/tz; or for some erosion parameters as "sedporos", "RohSedCH", "RohFluidCH");
- Configurable multi-layer for glacier, soil and groundwater representation;
- New trapezoidal section support for channels;
- Added classic RK solver to integrate non-linear dynamic models;
- Configurable d4 vs d8 cell architecture;
- win/mac/linux support;
- intel/gnu compiler support;
- Use as external library to call topkapi functionality from different language, as c++, python or Julia;
- 'plugin' support to nest external services as crop growth models (AquaCrop) or groundwater models (MODFLOW).

7.3.4 How to enhance maintainability

- Use of modern and uniform convention to assign name of variables, structures, procedures and modules;
- Homogeneous programming and comment style;
- Use of formal modules;
- Automatic api documentation (doxygen);
- Versioning (git);
- Logical separation between main process and specific processes (no calc code in the main);
- Module template definition to extend functionalities.