I found the paper significantly improved in the text and structure, however there may be some problems with the analysis, which I would like you to check before I proceed further. Below my notes while I was reading the paper.

Abstract.

Page1, Line 1: The first sentence says that “understanding”…”is rarely fully understood”. Please rephrase.

Page1, Line 12: 12…166 → 12-166

Page1, Line 15: finding are → finding is

Page1, Clarify whether forward and backward are your own definitions or from the cited papers.

Page 5, Line 15: How do you calculate the water balance error? What is the reference?

Page 6, Line 17: You say that you use Larsim to use areal estimates of P, but then in line 20 you say that P is the model forcing.

Page 7, Line 4: easy to → easy way to

Equation B3 should look like:

$$\int_{t_0}^{t^*} Pdt\left(\frac{S\left(t^*\right)}{S_{\max}}\right)^{\beta} - \int_{t_0}^{t^*}\left[\int_{t_0}^{t} Pdt\beta\left(\frac{S\left(t\right)}{S_{\max}}\right)^{\beta-1}\frac{1}{S_{\max}}\frac{dS}{dt}\right]dt \tag{1}$$

There are two main differences from your formulation.

First, $uv\big|_{to}^{t*} = u\left(t^*\right)v\left(t^*\right) - u\left(t^0\right)v\left(t^0\right) \neq \left(u\left(t^*\right) - u\left(t^0\right)\right)\left(v\left(t^*\right) - v\left(t^0\right)\right)$

Note that $u\left(t^0\right)v\left(t^0\right) = 0$

Second, in the last term of the equation, you have taken $\int_{t_0}^{t} Pdt$ out of the integral. This cannot be done, because $\int_{t_0}^{t} Pdt$ is a function of time.

I made a simple Matlab example (see below) to show the difference in results between Equation 1 and the one you have in your paper. To simplify the calculation, I assumed beta=1. You can see that Equation 1 gives the right results, returning cumsum(Q), whereas your equation provides a result that differs significantly from cumsum(Q).

The problem in Equation B3 propagates through the whole paper, and strongly affects your results.

I recommend checking this (and the rest of the equations) before I proceed in further reviewing the paper.

```
clear all
close all
% Fabrizio Fenicia, review of Simon Seibert et al, hess-2016-109
% Matlab code

P=[0;2;5;1;0;20;0];
Smax=50;
% beta=1;

nT=length(P);
S=NaN(1,nT);
Q=NaN(1,nT);
delT=1;

Sst=20;
S(1)=Sst;

% storage and flux from HBV model, using fixed step implicit approximation
for i=1:nT
    S(i)=Smax*(delT*P(i)+S(i))/(delT*P(i)+Smax);
    Q(i)=P(i)*(S(i)/Smax);
    if i<nT
        S(i+1)=S(i);
    end
end

figure
t=1:nT;
hold on
plot(t,S,'r');
plot(t,Q,'b');

sumQ=sum(Q);

cumP=sum(P)*delT;

% My implementation
add1_a=cumP*S(nT)/Smax;
cumSumP=cumsum(P);
dSdT=S(2:end)-S(1:end-1);
dSdT=[dSdT 0];

add2_int=cumSumP(:).*dSdT(:)/Smax;
add2_a=sum(add2_int)*delT;

sumQ_est_a=add1_a-add2_a;
```

```
res_a=sumQ_est_a-sumQ;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Your implementation

add1_b=cumP*(S(nT)-Sst)/Smax;
add2_int=sum(dSdT(:)/Smax)*delT;
add2_b=cumP*add2_int;

sumQ_est_b=add1_b-add2_b;

res_b=sumQ_est_b-sumQ;
```