# APPROXIMATE SOLUTION TO THE AUTOCATALYTIC HYDROLYSIS OF CELLULOSE

$$C + H \xrightarrow{k} P + H$$

$$P \xrightarrow{k_a} A + \alpha H$$

$$\frac{dC}{dt} = -kHC$$

$$\frac{dP}{dt} = kHC - k_a P$$

$$\frac{dH}{dt} = \alpha k_a P$$

$C(0) = C_0$, $P(0) = 0$, and $H(0) = H_0$

$$C + P + H/\alpha = C_0 + H_0/\alpha$$

Gehlen, M.H. *Approximate solution of the autocatalytic hydrolysis of cellulose*, Cellulose **2009**, 16, 1069 – 1073.
Gehlen, M.H. *Kinetics of autocatalytic acid hydrolysis of cellulose with crystalline and amorphous fractions*, Cellulose, **2010**, 17, 245 – 252.

$$C(t) = C_0 \exp\left[ - k \int_0^t H(\tau) d\tau \right]$$

$$H(t) = H_0 + \alpha k_a \int_0^t P(\tau) d\tau$$

First-order approximation

1) $H(t) = H_0$

$$C_1(t) = C_0 \exp[- kH_0 t]$$

$$P_1(t) = k \exp[- k_a t] \otimes H_0 C_1(t)$$

$$P_1(t) = \frac{C_0 k H_0}{(kH_0 - k_a)} \left( \exp[- k_a t] - \exp[- kH_0 t] \right)$$

## Second-order approximation: $kH_0 > k_a$

$$H_1(t) = H_0 + \frac{\alpha C_0}{(kH_0 - k_a)}\left(kH_0\left(1 - \exp\left[-k_a t\right]\right) - k_a\left(1 - \exp\left[-kH_0 t\right]\right)\right) \qquad C_1(t) = C_0 \exp\left[-kH_0 t\right]$$

$$\ln\frac{C_2(t)}{C_1(t)} = \frac{-\alpha k C_0}{(kH_0 - k_a)}\left(\left(kH_0 - k_a\right)t + \frac{kH_0}{k_a}\left(\exp\left[-k_a t\right] - 1\right) - \frac{k_a}{kH_0}\left(\exp\left[-kH_0 t\right] - 1\right)\right)$$

Approximate solution found has twice the term $\exp[\exp[-\tau]]$, which is a classical Gompertz function used in ageing theory (Hallén 2007), as well as in modeling tumor growth (Laird 1964). Moreover, the Gompertz function has been used to describe the biodegradation of cellulose (Hu et al. 2004).

## Short time and long time behaviors

$$\ln\frac{C_2(t)}{C_1(t)} \cong \frac{-\alpha k_a k^2 H_0 C_0}{6} t^3$$

$$\ln\frac{C_2(t)}{C_1(t)} = \alpha C_0\left(\left(\frac{k}{k_a} + \frac{1}{H_0}\right) - kt\right)$$

# NUMBER OF SCISSIONS PER CELLULOSE CHAIN S(t)

$k_a >> kH_0$

$$S(t) = \frac{C_0(1-y)}{(1+y\alpha C_0/H_0)}$$

$$y = \exp[-k(H_0 + \alpha C_0)t]$$

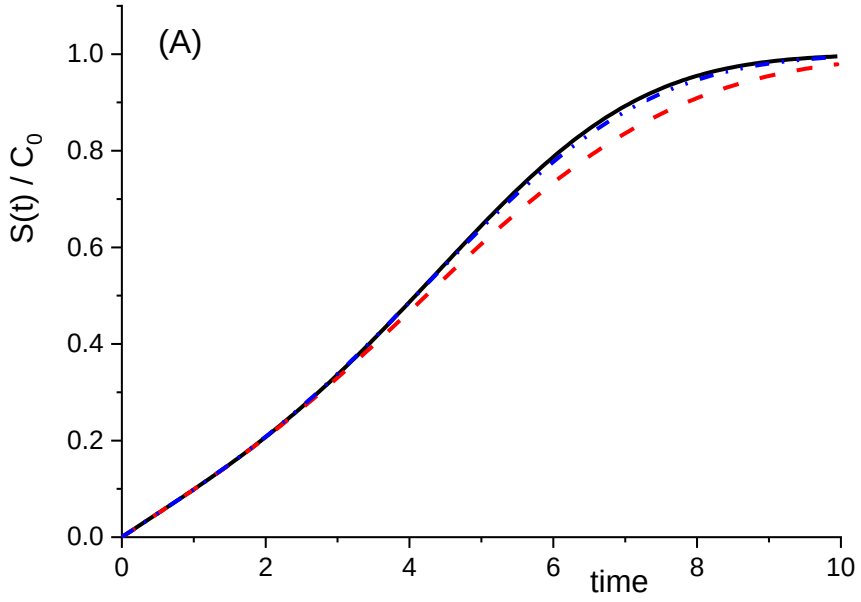
Classical sigmoid shape of logistic function is recovery by S(t)

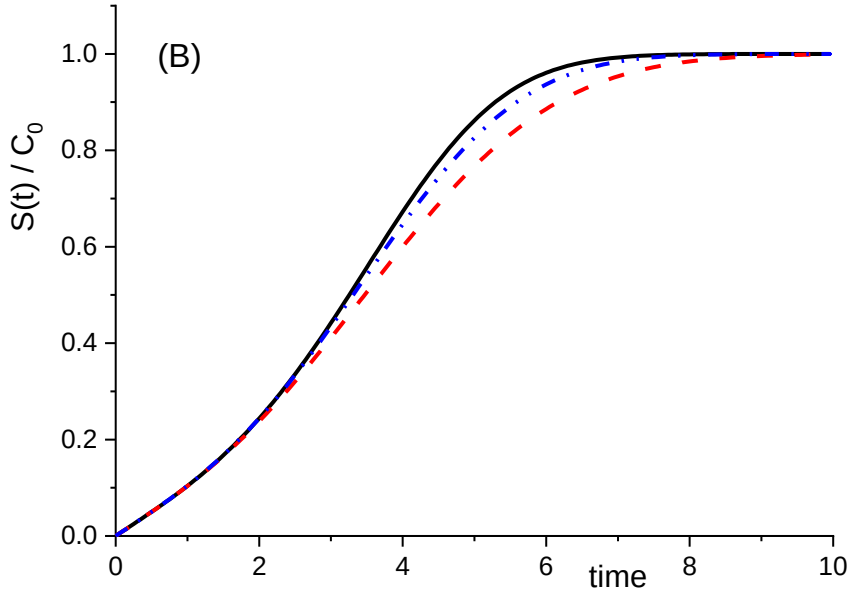
# NUMBER OF SCISSIONS PER CELLULOSE CHAIN S(t)

$$S(t)/C_0 = 1 - C(t)/C_0$$

$$\ln\frac{C_2(t)}{C_1(t)} = \frac{-\alpha k C_0}{(kH_0 - k_a)}\left( (kH_0 - k_a)\,t + \frac{kH_0}{k_a}\left(\exp[-k_a t] - 1\right) - \frac{k_a}{kH_0}\left(\exp[-kH_0 t] - 1\right)\right)$$

$$C_1(t) = C_0 \exp[-kH_0 t]$$



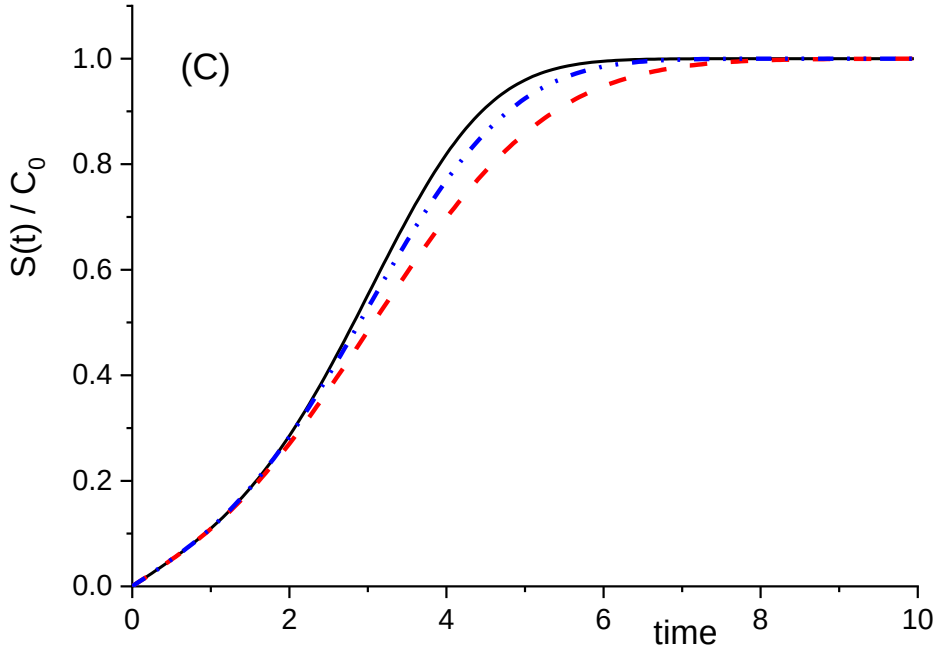




**Figure 1**. Time evolution of the normalized number of scissions per cellulose chain. Continuous line is the numerical result while (– – –) is the $C_2(t)$ approximation and (– -- –) is the cubic expansion given by eq. 13. Constant parameters are $H_0 = 1.0$; $C_0 = 50$; $k = 0.1$ and $\alpha = 1$. The rate constant $k_a$ is changed from 0.05 (A); 0.12 (B) and 0.2 (C).

Surprising result: Cubic expansion is very good!!!

$$C(t) = C_0 \exp\left[ -kH_0 t - \frac{\alpha k_a k^2 H_0 C_0}{6} t^3 \right]$$

## Life-time ($t_{1/2}$): $C_2(t) = C_0/2$

$$t_{1/2}^3 + \frac{6}{\alpha k_a k C_0} t_{1/2} = \frac{6\ln 2}{\alpha k_a k^2 C_0 H_0}$$

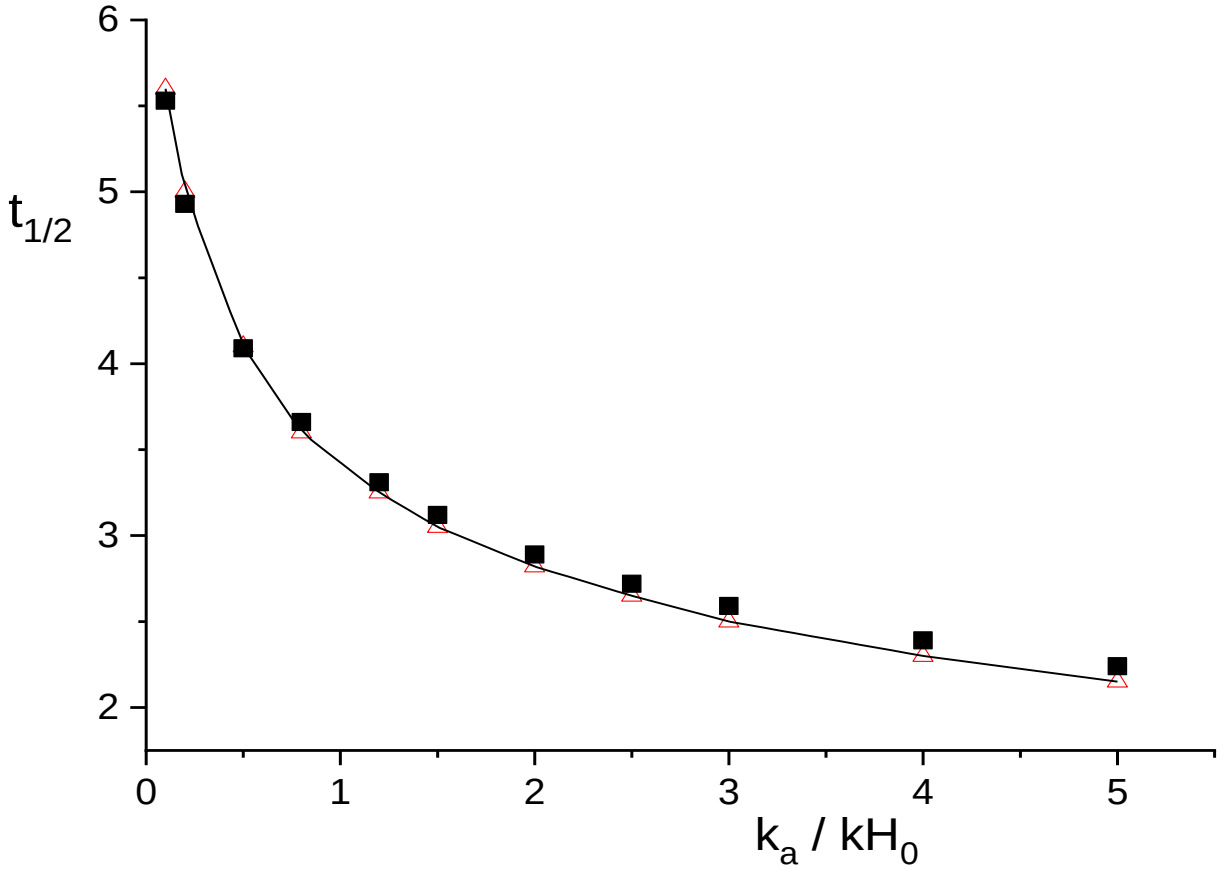
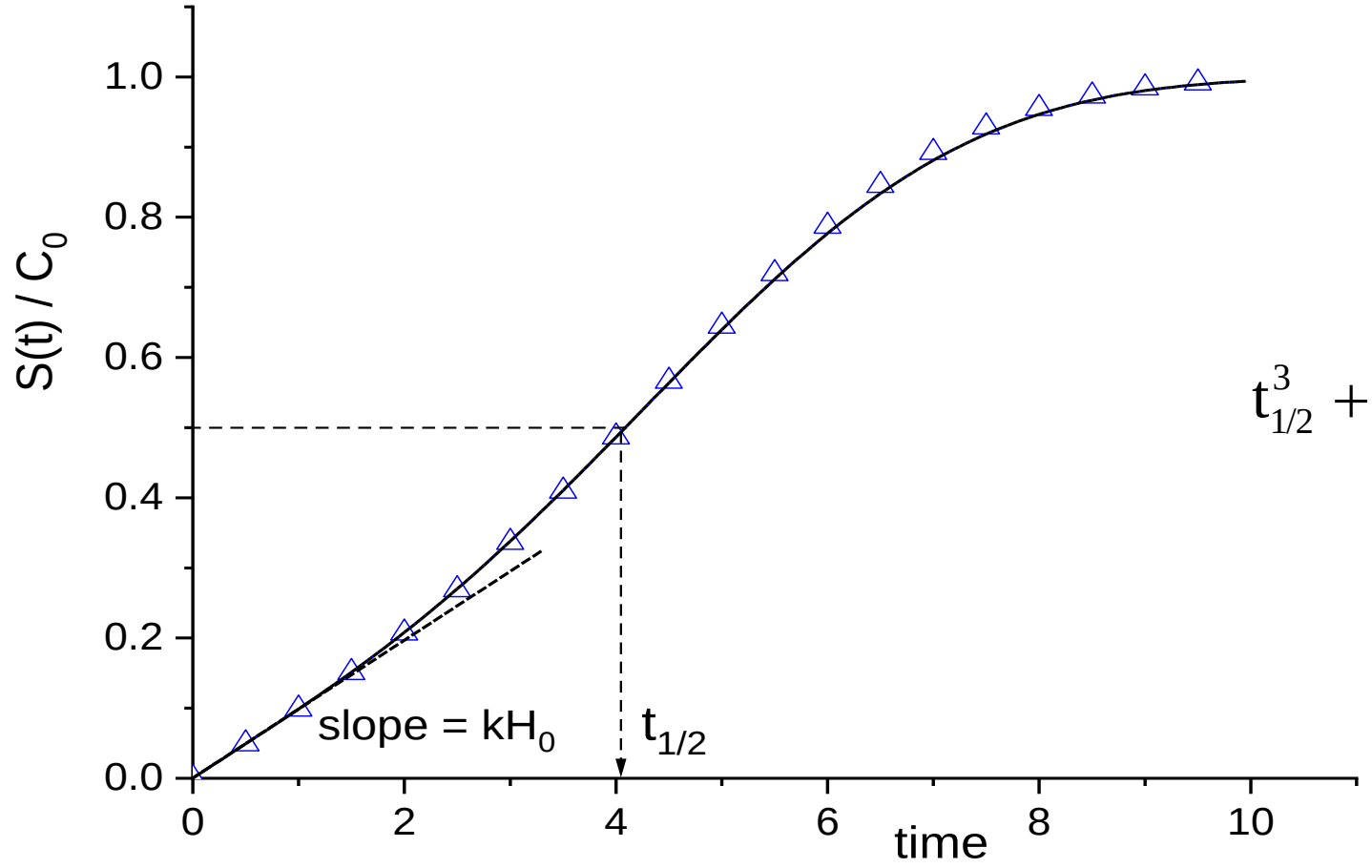



**Figure 2.** Lifetime as a function of the ratio $k_a/kH_0$. Numerical result (Δ) and value of lifetime $t_{1/2}$ calculated using the derived equation (■).

Experimental Protocol




$$t_{1/2}^3 + \frac{6}{\alpha k_a k C_0} t_{1/2} = \frac{6\ln 2}{\alpha k_a k^2 C_0 H_0}$$

**Figure 3**. Scheme of determination of the rate constant parameters from analysis of the initial slope and value of lifetime from the time evolution of $S(t)/C_0$. Numerical result ($\Delta$) and continuous line is the cubic expansion given by Eq. 13. Simulated parameters used are $H_0 = 1.0$; $C_0 = 50$; $k = 0.1$; $\alpha = 1$; and $k_a = 0.05$; $t_{1/2} = 4.08$.

```
c             PROGRAM cell_v2.for
c     This program calculates the relative concentration
c     of celulose from a autocatalytic model and performs comparison
c     with second order approximation, and its cubic function.
c     It also calculates de lifetime from cubic approximation function
c     Author: Marcelo H. Gehlen
c     Last version: february 2009
c     Driver for routine RKDUMB
      real ss,tt,rql,pl,p2,p0
        common /PATH/X(500),Y1(3,500)
        common /PARAM/ah0,alfa,c0,rk,ak
        parameter(NVAR=3,NSTEP=400)
        dimension VSTART(NVAR)
      dimension sn(2000)
      external DERIVS
        character fname*64
        fname='cel   .dat'
        write(*,*)'enter 3 digits file identifier'
        read(*,*)id
        write(fname(4:6),5)id
5       format(i3.3)
        open(unit=1,file=fname,form='formatted',
     $ access='sequential')
6     format(f10.4,f10.4)
7     format(f10.4,f10.4,f10.4)
8       format(f10.4,f10.4,f10.4,f10.4)
      write(*,*)'initial H0 mM'
        read(*,*)ah0
      write(*,*)'alfa'
      read(*,*)alfa
      write(*,*)'cellulose C0'
      read(*,*)c0
        write(*,*)'Degraded rate constant'
        read(*,*)rk
      write(*,*)'acid catalysis rate constant'
      read(*,*)ak
      write(*,*)'initial degrated p0'
      read(*,*)p0
        write(1,6)ah0,c0
      write(1,6)alfa,p0
      write(1,6)rk,ak
c     Evaluation of the lifetime in cubic approximation
      xl=0.0
      rl=ah0*rk
      drk=ak-rl
      x2=100.0*rl
      aql=alfa*ak*rk*c0
      aq=2.0/aql
      r=2.07944/(aql*rl)
      tl2=0.69315/rl
      aq3=aq**3.0
      r2=r**2.0
      D=real(aq3+r2)
      rql=SQRT(D)
      rq2=real(rql)
      sl=r+rq2
      ss=sl**(1.0/3.0)
      rd=r-rq2
      rdl=abs(rd)
      tt=-(rdl)**(1.0/3.0)
      tl2cell=ss+tt
      print*,ss,tt,tl2cell
      pause
      write(1,7)xl,tl2,tl2cell
```

```
      VSTART(1)=c0
    VSTART(2)=p0
    VSTART(3)=ah0
      CALL RKDUMB(VSTART,NVAR,X1,X2,NSTEP,DERIVS)
    do 110 j=1,nstep,2
        sn(j)=1.0-y1(1,j)/c0
      t=x(j)
        p1=exp(-rl*t)-1.0
      p2=exp(-ak*t)-1.0
        fal=alfa*rk*c0*(drk*t+ak*p1/rl-rl*p2/ak)/drk
      c1=c0*exp(-rl*t)
        c2=c1*exp(-fal)
      sn2=1.0-c2/c0
c     Evaluation of the cubic approximation function
        arg3=aql*rl*t**3.0
      arg=arg3/6.0
      c3=c1*exp(-arg)
      sn3=1.0-c3/c0
      write(1,8)x(j),sn(j),sn2,sn3
110   continue
    close(unit=1)
    stop
      end
      SUBROUTINE DERIVS(X,Y,DYDX)
      common/PARAM/ah0,alfa,c0,rk,ak
      dimension y(3),dydx(3)
      dydx(1)=-rk*y(3)*y(1)
    dydx(2)=rk*y(3)*y(1)-ak*y(2)
    dydx(3)=alfa*ak*y(2)
      return
    end
```

```
      SUBROUTINE RKDUMB(VSTART,NVAR,X1,X2,NSTEP,DERIVS)
      PARAMETER (NMAX=3,NSTPMX=500)
      COMMON /PATH/ XX(NSTPMX),Y1(NMAX,NSTPMX)
      DIMENSION VSTART(NVAR),V(NMAX),DV(NMAX)
      DO 11 I=1,NVAR
        V(I)=VSTART(I)
        Y1(I,1)=V(I)
11    CONTINUE
      XX(1)=X1
      X=X1
      H=(X2-X1)/NSTEP
      DO 13 K=1,NSTEP
        CALL DERIVS(X,V,DV)
        CALL RK4(V,DV,NVAR,X,H,V)
        IF(X+H.EQ.X)PAUSE 'Stepsize not significant in RKDUMB.'
        X=X+H
        XX(K+1)=X
        DO 12 I=1,NVAR
          Y1(I,K+1)=V(I)
12      CONTINUE
13    CONTINUE
      RETURN
      END
      SUBROUTINE RK4(Y,DYDX,N,X,H,YOUT)
      PARAMETER (NMAX=10)
      DIMENSION Y(N),DYDX(N),YOUT(N),YT(NMAX),DYT(NMAX),DYM(NMAX)
      HH=H*0.5
      H6=H/6.
      XH=X+HH
      DO 11 I=1,N
        YT(I)=Y(I)+HH*DYDX(I)
11    CONTINUE
      CALL DERIVS(XH,YT,DYT)
      DO 12 I=1,N
        YT(I)=Y(I)+HH*DYT(I)
12    CONTINUE
      CALL DERIVS(XH,YT,DYM)
      DO 13 I=1,N
        YT(I)=Y(I)+H*DYM(I)
        DYM(I)=DYT(I)+DYM(I)
13    CONTINUE
      CALL DERIVS(X+H,YT,DYT)
      DO 14 I=1,N
        YOUT(I)=Y(I)+H6*(DYDX(I)+DYT(I)+2.*DYM(I))
14    CONTINUE
      RETURN
      END
```