Stability Properties of some Algorithms

for the solution of Non-linear

Dynamic Vibration Equations

by

W.L. Wood and M.E. Oduor

Department of Mathematics

University of Reading

SUMMARY

This paper describes numerical experiments made to investigate
the stability of some time-stepping algorithms applied to the equation
$\ddot{u} + P(u) = 0$ representing a non-linear elastic spring. These
algorithms would be unconditionally stable when applied to linear
problems but here they may be only conditionally stable. Ways of
improving the stability are demonstrated; the effect of linearisation
is also investigated.

INTRODUCTION

This paper describes numerical experiments made to investigate the stability of some time-stepping algorithms applied to non-linear equations of the form $\ddot{u} + P(u) = 0$:

$$\ddot{u} + S_1 u(1 + S_2 u^2) = 0 , \qquad S_1, S_2 > 0 \qquad (1)$$

which represents a non-linear elastic hardening spring and

$$\ddot{u} + S \tanh(u) = 0 , \qquad S > 0 \qquad (2)$$

which represents a non-linear elastic softening spring.

The inspiration for this work comes from a paper by Park [1] who gives the unsatisfactory behaviour of the trapezium rule/Newmark [2] average acceleration method when applied to equations (1) and (2) as the motivation for introducing the Park 3-step scheme (which is a linear combination of the Gear 2-step and 3-step methods [3]). We record here the results of applying to equations (1) and (2) the Newmark algorithm in its single-step form and the single-step SS22 algorithm [4] in three different forms of which one is linearised. The particular merit of single-step algorithms is the ease of changing the time step size.

BACKGROUND THEORY

We first note that equations (1) and (2) are examples of the conservative system

$$\ddot{u} + P(u) = 0 \qquad (3)$$

which represents a constant energy situation:

$$\tfrac{1}{2}\dot{u}^2 + \int P(u)\,du = \text{constant}. \tag{4}$$

For equation (1) we have

$$\tfrac{1}{2}\dot{u}^2 + S_1\frac{u^2}{2} + S_1 S_2\frac{u^4}{4} = \tfrac{1}{2}v_0^2 + S_1\frac{u_0^2}{2} + S_1 S_2\frac{u_0^4}{4} \tag{5}$$

where u_0, v_0 are the values of $u(0)$, $\dot{u}(0)$ respectively.

For equation (2) we have

$$\tfrac{1}{2}\dot{u}^2 + S\ln(\cosh(u)) = \tfrac{1}{2}v_0^2 + S\ln(\cosh(u_0)) \tag{6}$$

The solutions are periodic. If $v_0 = 0$ the amplitude is u_0.

Equation (1) has an exact solution in terms of Jacobian elliptic functions. If $v_0 = 0$ we have

$$u(t) = u_0\,\text{cn}(-\hat{\omega}t, k) \tag{7}$$

where

$$\hat{\omega}^2 = S_1(1 + S_2 u_0^2)$$

and

$$k^2 = \frac{u_0^2 S_2}{2(1 + u_0^2 S_2)}. \tag{8}$$

The period $\qquad T = 4K/\hat{\omega}$ (9)

where K can be obtained from Abramowitz and Stegun [5]. This is a useful check on the numerical results.

The period for the solution of equation (2) has to be measured from the converged solution for large amplitude but for u remaining

small enough for

$$\tanh(u) \approx u - \frac{u^3}{3} \tag{10}$$

we have the solution of

$$\ddot{u} + S(u - \frac{u^3}{3}) = 0 \tag{11}$$

given by

$$u(t) = u_0 \text{sn}(\hat{\omega}t + K, k) \tag{12}$$

where

$$v_0 = 0 ,$$

$$k^2 = u_0^2/(6 - u_0^2) \tag{13}$$

$(u_0 < \sqrt{3}$ is the condition for a periodic solution) and

$$\hat{\omega}^2 = \frac{1}{6}S(6 - u_0^2) .$$

The period is $T = 4K/\hat{\omega}$ and K is obtained from reference [5].

THE ALGORITHMS

The Newmark algorithm is implemented by solving for $\ddot{u}_{n+1}$ the non-linear equation

$$\ddot{u}_{n+1} + P(\tilde{u}_n + \Delta t^2 \beta \ddot{u}_{n+1}) = 0 \tag{14}$$

which represents equation (3) satisfied at $t = (n+1)\Delta t$ where

$$\tilde{u}_n = u_n + \Delta t \dot{u}_n + \frac{\Delta t^2}{2}(1 - 2\beta)\ddot{u}_n \tag{15}$$

is known from the values at the beginning of the time-step.

Then
$$u_{n+1} = \tilde{u}_n + \Delta t^2 \beta \ddot{u}_{n+1} \tag{16}$$

$$\dot{u}_{n+1} = \dot{u}_n + (1-\gamma)\Delta t \ddot{u}_n + \gamma \Delta t \ddot{u}_{n+1} \tag{17}$$

where β, γ are the standard Newmark parameters and Δt is the time step.

The SS22 algorithm is implemented by solving for α_n the weighted residual equation

$$\int_0^{\Delta t} \omega(\tau)\left[\alpha_n + P(u_n + \tau\dot{u}_n + \frac{\tau^2}{2}\alpha_n)\right]d\tau = 0 \tag{18}$$

where $\omega(\tau)$ is the 'weight'. Thus in SS22 the non-linear equation is satisfied in some average sense over the time interval. We want to introduce parameters θ_m where

$$(\Delta t)^m \theta_m = \int_0^{\Delta t} \omega(\tau)\tau^m d\tau / \int_0^{\Delta t} \omega(\tau)d\tau \quad . \tag{19}$$

Hence we have to decide how to expand the non-linear term. Here we take

$$P(u_n + \tau\dot{u}_n + \frac{\tau^2}{2}\alpha_n) = P(u_n) + (\tau\dot{u}_n + \frac{\tau^2}{2}\alpha_n)P'(\hat{u}_n) \quad . \tag{20}$$

Then equation (18) gives

$$\alpha_n + P(u_n) + \left[\Delta t \theta_1 \dot{u}_n + \frac{\Delta t^2}{2}\theta_2\alpha_n\right]P(\hat{u}_n) = 0 \tag{21}$$

as the equation to be solved for α_n. We now have to choose what to substitute for $\hat{u}_n$ in what is effectively the remainder term of the Taylor expansion in equation (20):

(i) $\hat{u}_n = u_n$ gives the linearised form

(ii) $\hat{u}_n = \frac{1}{2}(u_n + u_{n+1})$ is recommended in reference [4].

(iii) $\hat{u}_n = u_{n+1}$ gives a more 'implicit' form

also tested here to see what effect it has.

(ii) and (iii) give a non-linear equation to be solved

at each time step for α_n , with

$$u_{n+1} = u_n + \Delta t \dot{u}_n + \frac{\Delta t^2}{2} \alpha_n . \qquad (22)$$

The SS22 algorithm is completed by taking

$$\dot{u}_{n+1} = \dot{u}_n + \Delta t \alpha_n . \qquad (23)$$

For the hardening spring the corresponding non-linear equations

are solved using the NAG subroutine C02AEF and for the softening spring

the NAG transcendental function solver C05AJF is used. The latter

is an adaptation of Newton-Raphson.

NUMERICAL RESULTS

1. $P(u) = S_1 u(1 + S_2 u^2)$ (The hardening spring).

We take $S_1 = 100$, $S_2 = 10$ and starting conditions $v_0 = 0$

with $u_0 = 0.1, 1.5$ to give two cases with lesser and greater

non-linearity effects.

(a) Newmark

We take $\gamma = 0.5$, $\beta = 0.25$ which gives the trapezium rule/

average acceleration equivalent.

For both $u_0 = 0.1$ and $u_0 = 1.5$ this algorithm is unconditionally

stable and the numerical solution is not damped but as expected it

becomes distorted for values of the time step greater than about

T/10 where T is the period (for $u_0 = 0.1, 1.5$ the periods

are T = 0.61, 0.15 respectively to 2 figures from reference [6]).

Park refers to 'local instability' here but we find that the
numerical solution remains stable in the sense that the amplitude
(although inaccurate) remains bounded.   (Park's time scales
in his figures 8,9 should be multiplied by 10^{-1}).

(b)   SS22

(i) $\hat{u}_n = u_n$ (The linearised version). $\theta_1 = \theta_2 = 0.5$.

For both $u_0 = 0.1$ and $u_0 = 1.5$ this algorithm gives a stable
and undamped solution for smaller values of Δt but the damping
effect increases as Δt increases.   The algorithm is thus
unconditionally stable but the damping effect reduces the accuracy
for values of $\Delta t > T/10$.

(ii) $\hat{u}_n = \frac{1}{2}(u_n + u_{n+1})$, $\theta_1 = \theta_2 = 0.5$.

For $u_0 = 0.1$ this algorithm becomes unstable at about $\Delta t = 0.064$
which is approximately $T/10$.   Figure 1 shows the result with
this algorithm with $\Delta t = 0.1$ together with Newmark (a) with
the same time step.

For $u_0 = 1.5$ this algorithm becomes unstable at about $\Delta t = 0.01$
($T = 0.15$ here).   Experiments made to control the instability
by increasing the θ values are illustrated in Figure 2.   Taking
$\Delta t = 0.015$ this shows instability with $\theta_1 = \theta_2 = 0.5$ and
slight damping with $\theta_1 = \theta_2 = 0.55$.

(iii) $\hat{u}_n = u_{n+1}$, $\theta_1 = \theta_2 = 0.5$.

Figure 3 illustrates the damping effect produced by this algorithm
for $\Delta t = 0.1$ with $u_0 = 0.1$ (this would be unstable with (ii))
($T = 0.61$).

Figure 4 illustrates the effect produced by this algorithm with $u_0 = 1.5$ $(T = 0.15)$. Taking $\Delta t = 0.01$ gives a slightly damped solution. Taking $\Delta t = 0.03$ gives a solution which first increases in amplitude and then becomes steady with amplitude ≈ 2 . In the sense that the amplitude remains bounded, this is a stable solution.

2. $P(u) = S \tanh(u)$ (The softening spring).

We take $S = 100$ and starting conditions $v_0 = 0$ with $u_0 = 0.2, 4.0$ to give lesser and greater nonlinearity effects.

(a)   Newmark

We take $\gamma = 0.5$, $\beta = 0.25$.

For $u_0 = 0.2$ $(T \approx 0.63)$ this algorithm gives a stable solution up to the highest time step tested i.e. $\Delta t = 0.3$.

For $u_0 = 4.0$ $(T \approx 1.3)$ $\Delta t = 0.2$ is stable and $\Delta t = 0.3$ is unstable. This corresponds closely to Park's result. Park says he is using Newton-Raphson to solve the non-linear equation on each time step and we are using an adaptation of Newton-Raphson with the derivatives estimated.

(b)   SS22

(i) $\hat{u} = u_n$ (The linearised version).

For $u_0 = 0.2$ $(T \approx 0.63)$ the algorithm is stable for $\Delta t = 0.1$ and unstable for $\Delta t = 0.2$ (Figure 5).

For $u_0 = 4.0$ $(T \approx 1.3)$ $\Delta t = 0.03$ gives a solution which appears to be marginally stable. Figure 6 shows the result

with $\Delta t = 0.04$ which is definitely unstable, (to make Figure 6 clearer a reduced number of results are actually indicated).

Figure 7 shows the stabilising effect of increasing the θ values to $\theta_1 = \theta_2 = 0.55$ with $\Delta t = 0.1$.

(ii) $\hat{u} = \frac{1}{2}(u_n + u_{n+1})$, $\theta_1 = \theta_2 = 0.5$.

For $u_0 = 0.2$ this algorithm certainly gives a stable solution up to $\Delta t = 0.3$ $(T \approx 0.63)$.

For $u_0 = 4.0$ $(T \approx 1.3)$, Figure 6 also includes the solution with $\Delta t = 0.04$ which is slightly damped. As Δt is increased the solution becomes inaccurate and by $\Delta t = 0.2$ it is highly damped. At $\Delta t = 0.5$ the non-linear solver failed.

(iii) $\hat{u} = u_{n+1}$.

This algorithm gives conditional stability for both values of the amplitude. Figure 8 shows that for $u_0 = 0.2$ $(T \approx 0.63)$ the marginal value is around $\Delta t = 0.06$. The results with $\Delta t = 0.1$ are definitely unstable.

CONCLUSIONS

For the values of the parameters which give trapezium-rule-like versions of the algorithms i.e. (a) Newmark with $\gamma = 0.5$, $\beta = 0.25$ and (b)(ii) SS22 with $\theta_1 = \theta_2 = 0.5$ and $\hat{u} = \frac{1}{2}(u_n + u_{n+1})$ the first gives a better performance with the hardening spring and the second with the softening spring. We can always obtain stability by adjusting the parameters to induce artificial damping at the cost of decreasing the accuracy. Alternatively the 'rule of thumb' which

some engineers use (Reference 6 and personal communication) : $\Delta t \leqq T/100$ would certainly be safe with the cases considered here.

REFERENCES
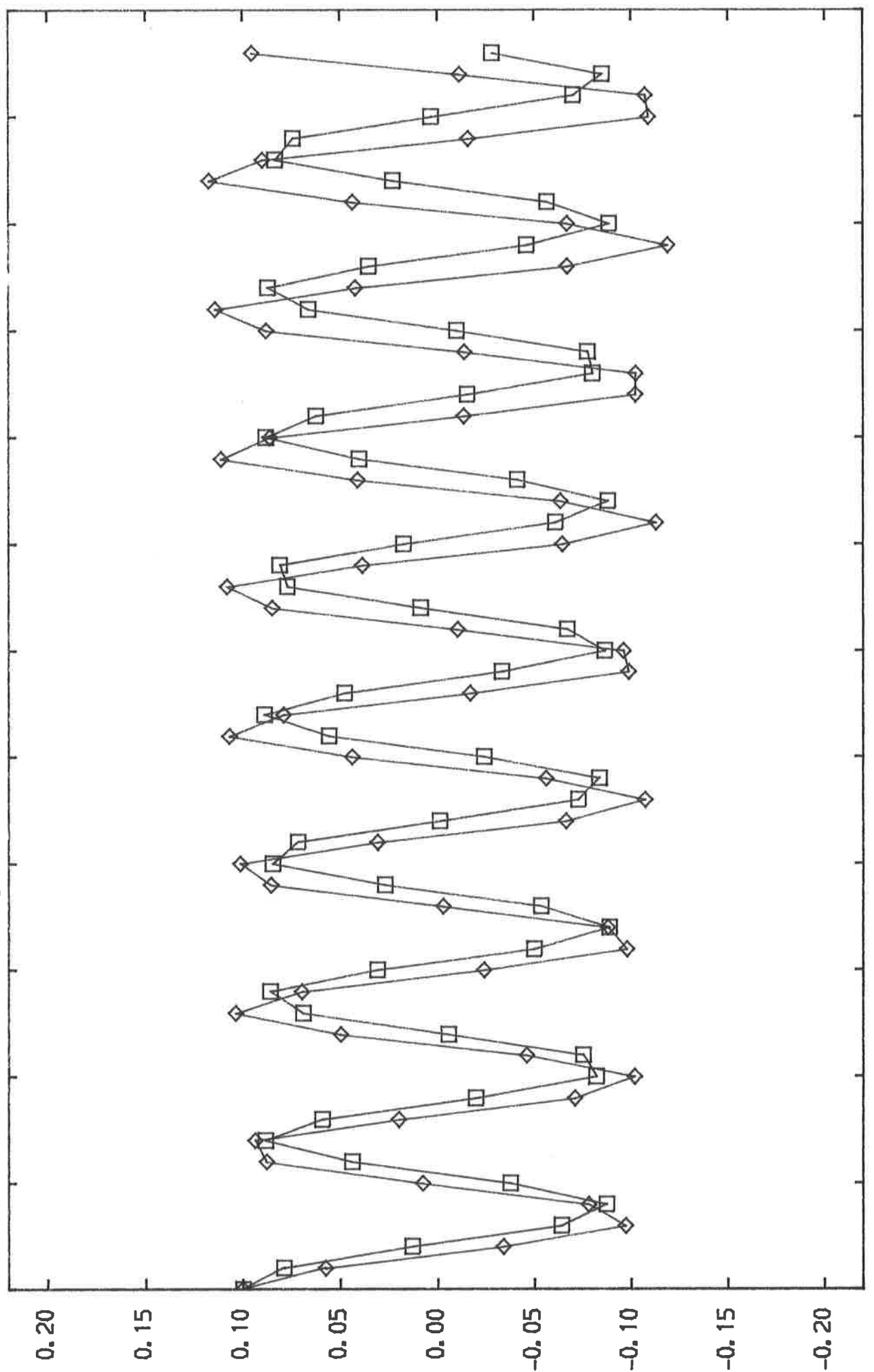
1. K.C. PARK.   An improved stiffly stable method for direct integration
            of non-linear structural dynamic equations.
            Trans. ASME. J.App.Mech. $\underline{42}$, 464-470   (1975).

2. N.M. NEWMARK.   A method of computation for structural dynamics.
            Proc.Am.Soc.Civ.Eng. $\underline{8}$, 67-94   (1959)

3. C.W. GEAR.   Numerical integration of stiff ordinary differential
            equations.
            Report no. 221, Dept. of Computer Science, Univ. of
            Illinois   (1967).

4. O.C. ZIENKIEWICZ, W.L. WOOD, N.W. HINE and R.L. TAYLOR.   A unified
            set of single-step algorithms Part 1.
            IJNME $\underline{20}$, 1529-1552   (1984).

5. M. ABRAMOWITZ and I.A. STEGUN.   Handbook of Mathematical Functions.
            National Bureau of Standards   App.Math.Series 55.
            Washington D.C.   (1964).

6. PAN CHANGSHI and YANG LI.   A preliminary analysis of the seismic
            responses of a railway tunnel.
            Int.Cong. on Large Underground Openings
            Florence June 1986.

LIST OF FIGURES

Figure 1

◇◇◇◇◇ (SS22-METHOD) ; CASE: W" +S1 (1+S2W**2) W=0 ◇◇◇◇◇

□□□□□ (NEWMARK-METHOD) ; CASE: W" +S1 (1+S2W**2) W=0 □□□□□
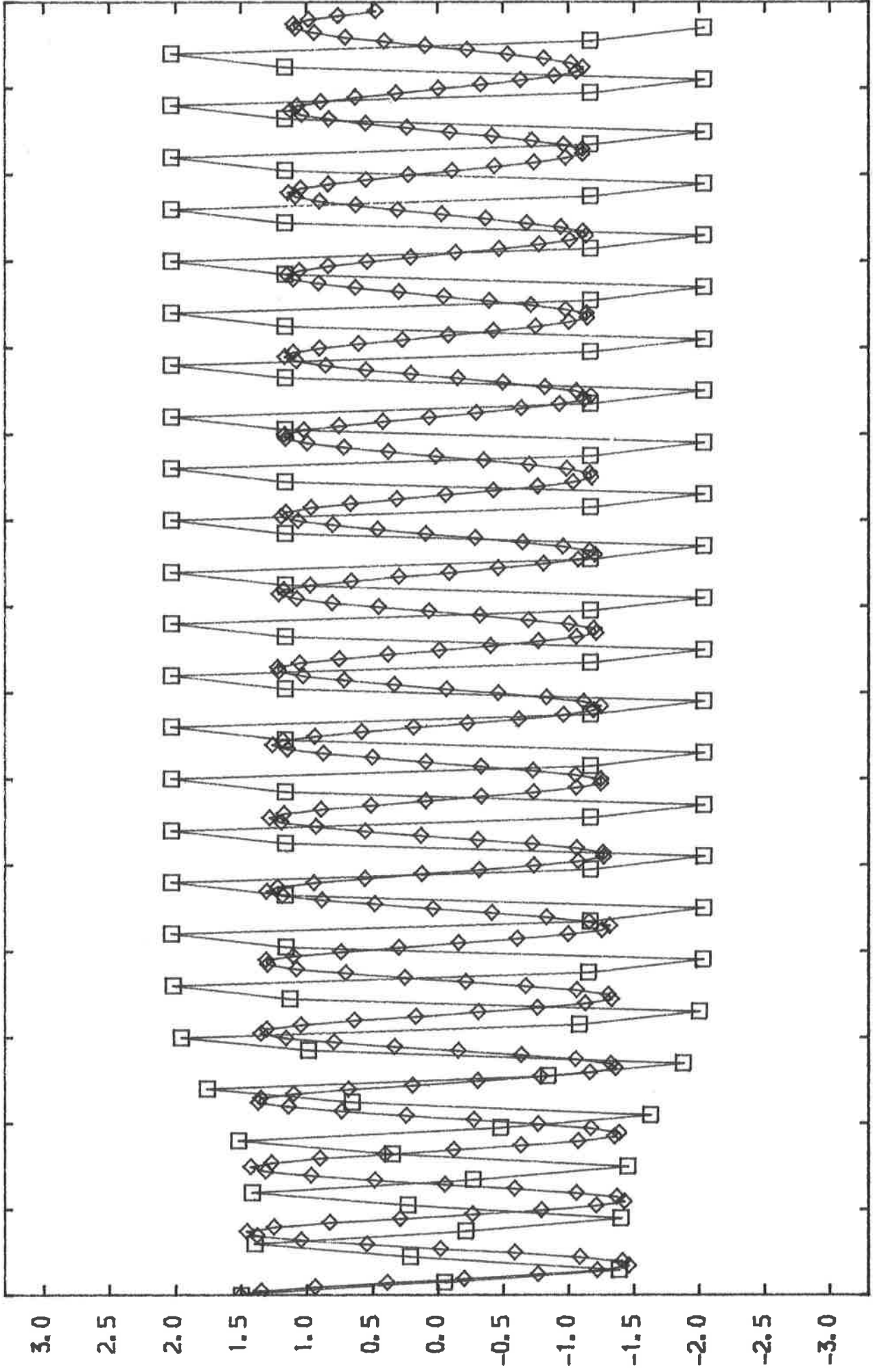
Figure 2

□□□□□ (SS22-METHOD) ; CASE: W"+S1 (1+S2W**2) W=0   □□□□□ $\vartheta_1 = \vartheta_2 = 0.5$

◇◇◇◇ (SS22-METHOD) ; CASE: W"+S1 (1+S2W**2) W=0   ◇◇◇◇ $\vartheta_1 = \vartheta_2 = 0.55$
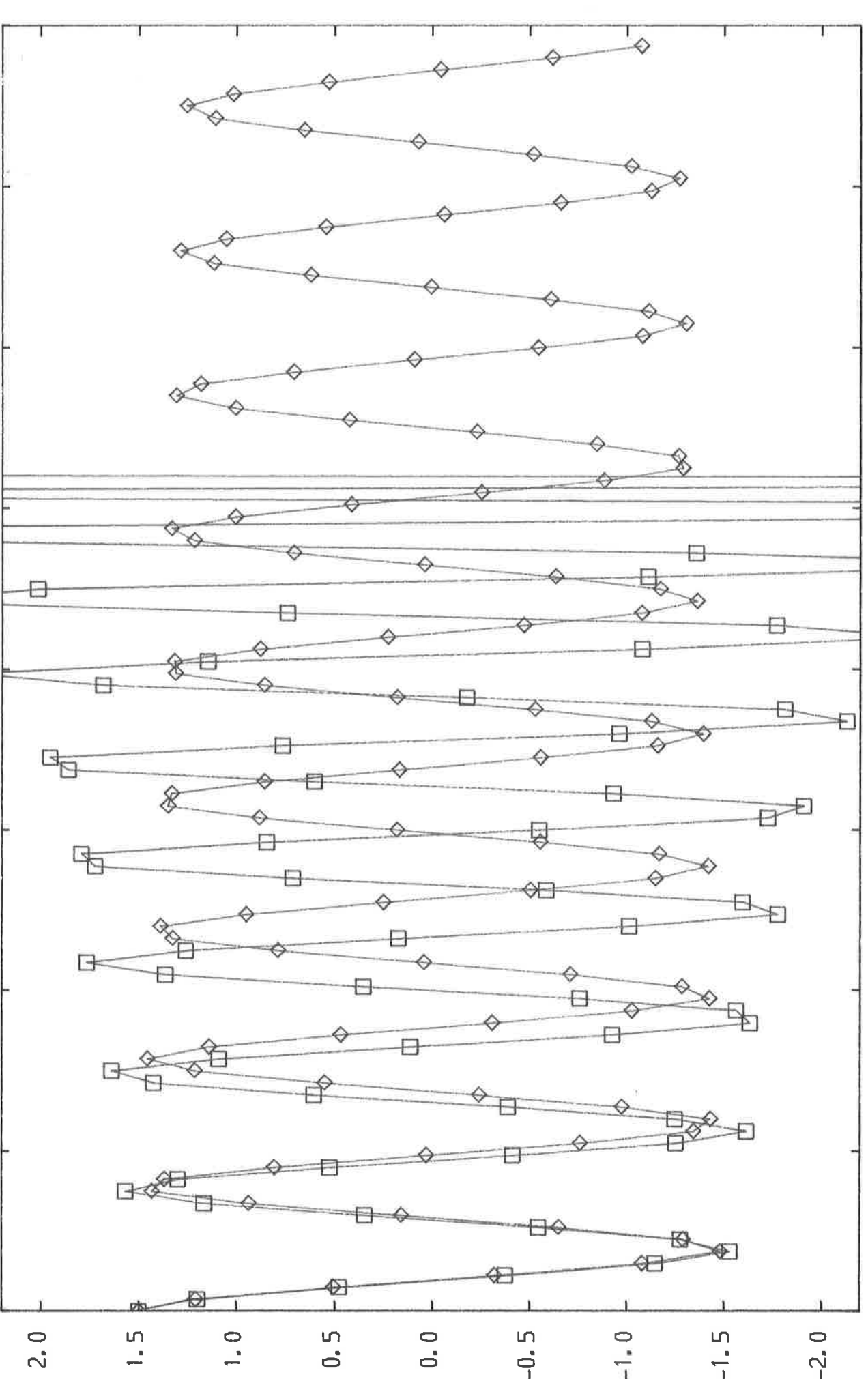
Figure 3

(SS22-METHOD) ;CASE; W" +S1 (1+S2W**2) W=0

Figure 4

◇◇◇◇◇ (SS22-METHOD) , CASE, W''+S1 (1+S2W**2) W=0   ◇◇◇◇◇ Δt = 0.01

□□□□□ (SS22-METHOD) , CASE, W''+S1 (1+S2W**2) W=0   □□□□□ Δt = 0.03

Figure 5

□□□□□ (SS22-METHOD) ; CASE, W" +S1TANH (W) =0   □□□□□ (ii)

◇◇◇◇◇ (SS22-METHOD) ; CASE, W" +S1TANH (W) =0   ◇◇◇◇◇ (i)
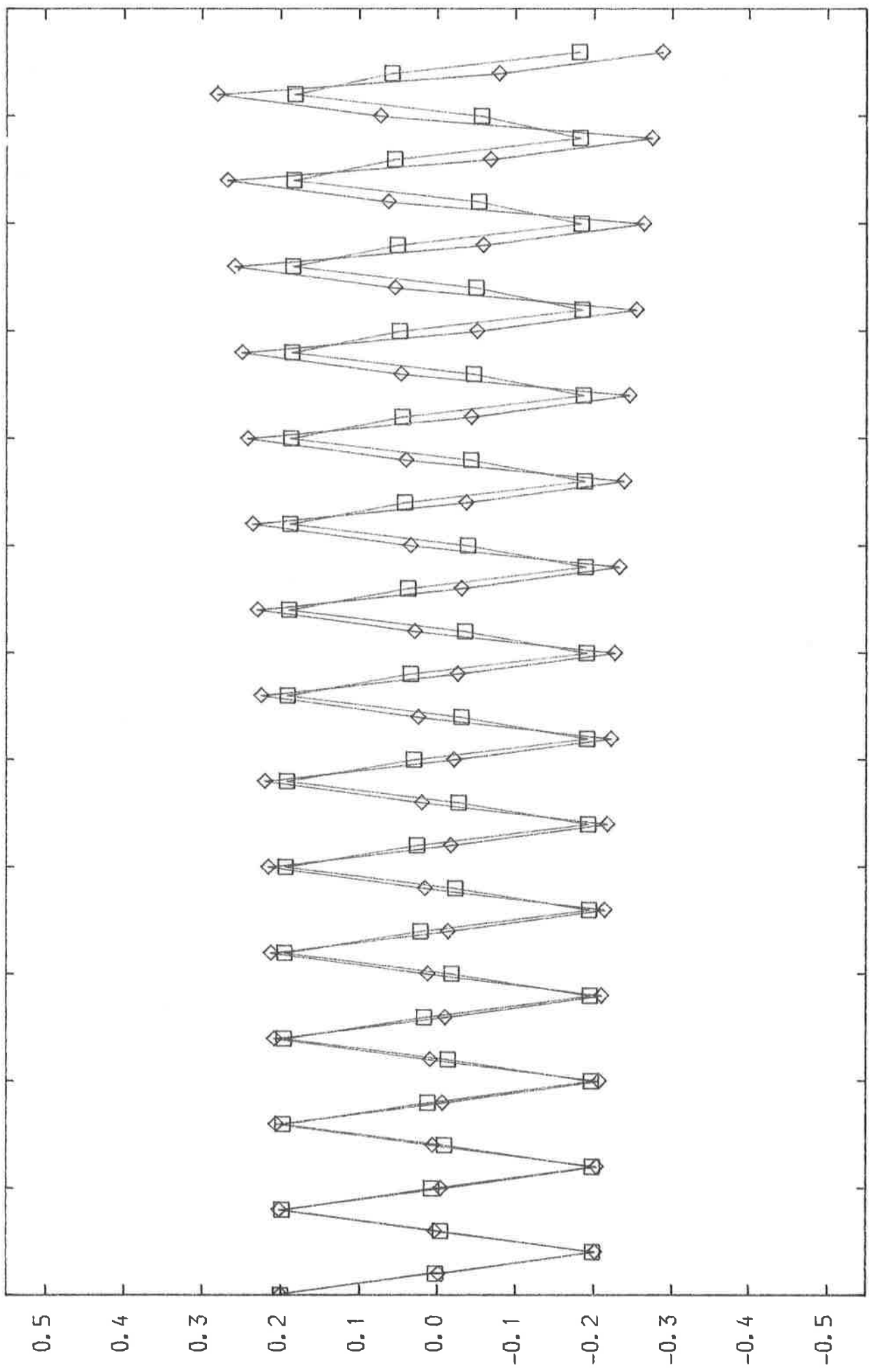
Figure 6

◇◇◇◇◇ (SS22-METHOD) ,CASE,W"+S1TANH(W) =0 ◇◇◇◇◇ (ii)
□□□□□ (SS22-METHOD) ,CASE,W"+S1TANH(W) =0 □□□□□ (i)
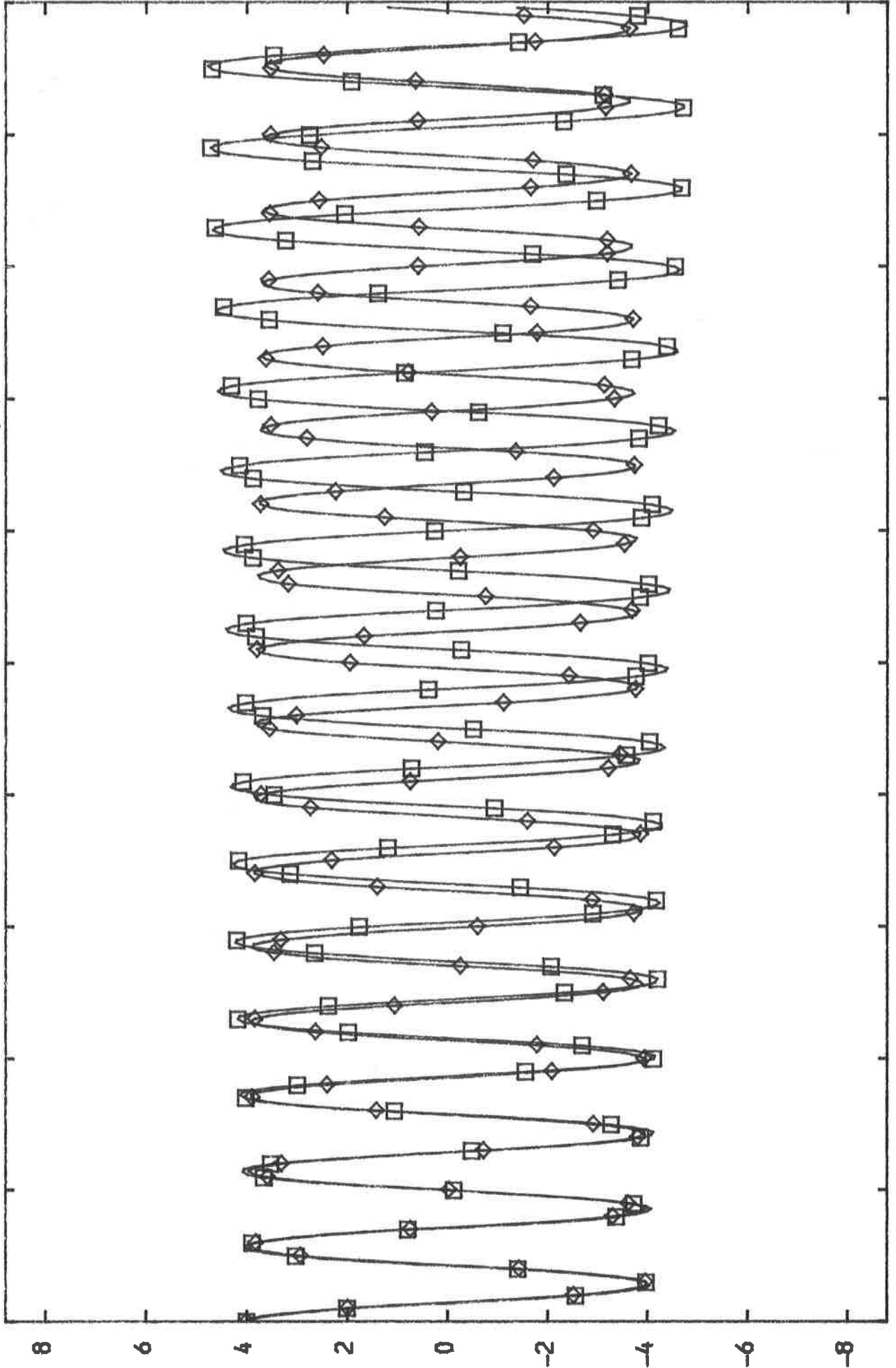
Figure 7

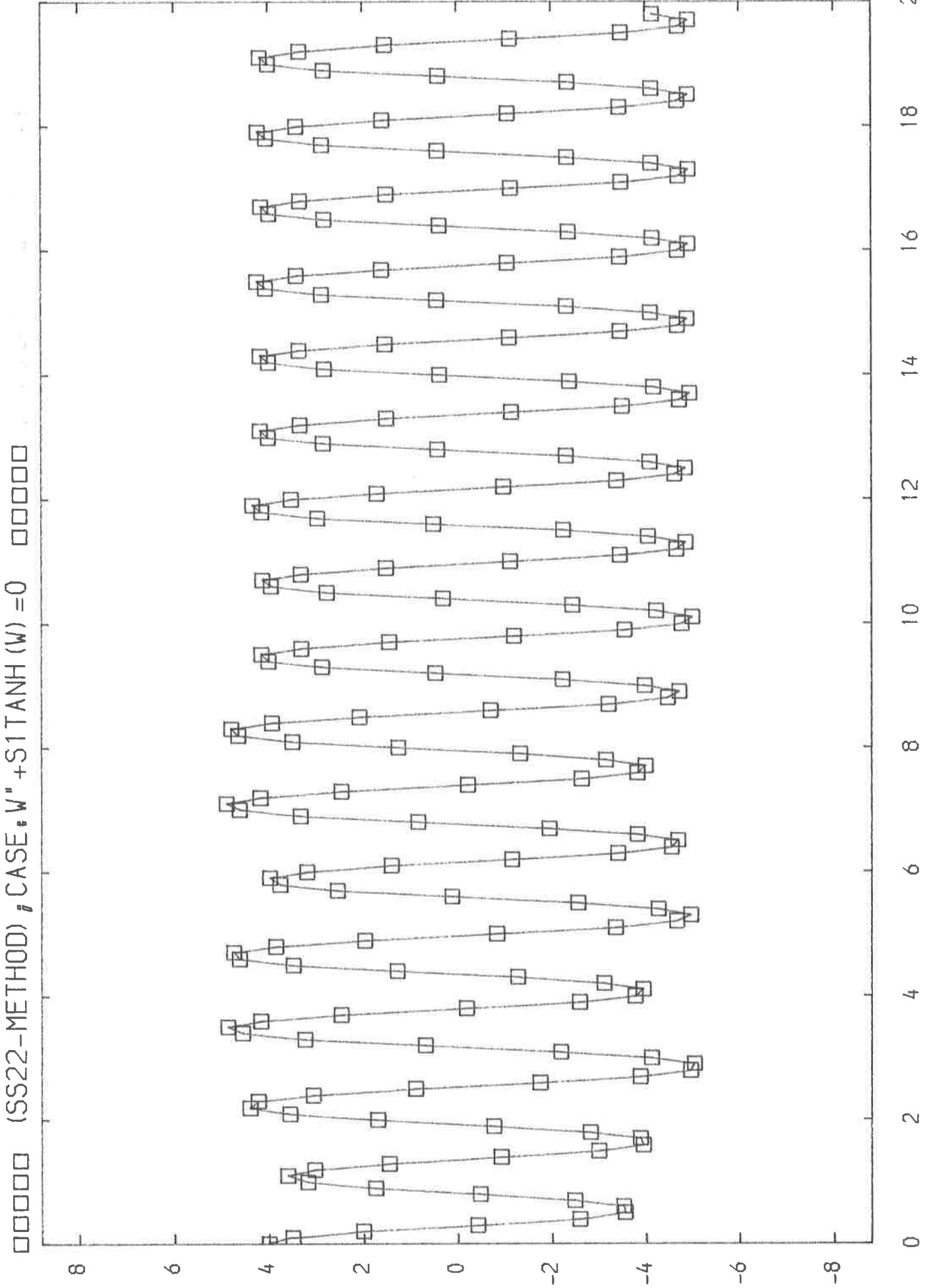□□□□□ (SS22-METHOD) ; CASE, W" +S1TANH (W) =0   □□□□□

Figure 8

◇◇◇◇◇ (SS22-METHOD) , CASE , W'' +S1TANH (W) =0   ◇◇◇◇◇   Δτ = 0.06

□□□□□ (SS22-METHOD) , CASE , W'' +S1TANH (W) =0   □□□□□   Δt = 0.1