N. Ramesh Babu, K.N. Murty*, V.V.S.S.S. Balaram
Aurora's Scientific, Technological and Research Academy
Department of Mathematics and Computer Science
(Bandlaguda, Hyderabad – 500 005, (A.P) India)

Control of Time Scale Dynamical Systems with an Application to Concurrency Control for Real-time Database Systems

(Recommended by Prof. E. Dshalalow)

Main objective in this paper is to unify results on controllability and observability on time scales and deduce the results of classical theory as a particular case and then resolve the time constraints on concurrency control by incorporating jump operators on time scale dynamical systems.

Обобщены результаты исследований управляемости и наблюдаемости при масштабировании во времени. Получены результаты классической теории как частного случая, при этом устранены ограничения по времени при параллельном управлении операторами перехода для динамических масштабируемых систем.

K e y w o r d s: time scale dynamical system, modern control system theory.

1. Introduction. Time scale dynamical system is an interesting area of current research and a great deal of work has been done by many authors in recent years [1]. From a modelling point of view, it is perhaps more realistic to model a real world phenomenon by a time scale dynamical system as it incorporates both continuous and discrete systems as a particular case [2, 3]. A fascinating fact is that all the widely different disciplines of application depend on a common core of Time scale dynamical system of the modern control system theory. These techniques require real time database systems that run effectively without any conflicts. In fact the concurrency control method receives certain information gathered from the transactions made in order to find and resolve conflicts [4]. Further, a real time transaction is a transaction with additional real-time attraction and importance.

Our main objective in this paper is to unify results on controllability and observability on time scales and deduce the results of classical theory as a particular case and then resolve the time constraints on concurrency control by incor-

* Corresponding author, email : nkanuri@hotmail.com

porating jump operators on time scale dynamical systems. More specifically, the paper is organized as follows: Section 2 presents some salient features of time scale dynamical systems that are needed for our later discussion. We introduce the concepts of controllability and observability for m-input, p-output, n-dimensional linear systems on time scales in section 3. In this section we present a set of necessary and sufficient conditions for the first order time scale dynamical system to be completely controllable and observable. For a complete theory on control on linear system we refer [5]. Section 4 is concerned with Real Time Database Systems and in fact deals with concurrency control of the database systems.

2. Basic results. In this section, we outline some of the basic notions concerning time scales. A time scale T is a closed subset of R; and examples of time scales include N, Z, R, cantor's set fuzzy sets etc. The set $Q = \{t \in R \setminus Q, 0 \le t \le 1\}$ are not time scales. Time scales need not necessarily be connected. In order to overcome this deficiency, we introduce the notion of jump operators. The mappings $\sigma, \rho : T \to T$ defined by

$$\sigma(t) = \inf\{s \in T : s > t\}, \ \rho(t) = \sup\{s \in T : s < t\},$$

are called jump operators.A point $t \in T$ is said to be right dense if $\sigma(t) > t$, left dense, if $\rho(t) = t$ and left scattered, if $\rho(t) < t$. The graininess $\mu : T \to [0, \infty)$ is defined by $\mu(t) = \sigma(t) - t$.

We say that f is rd-continuous if it is continuous in right dense points and if $\lim f(s)$ as $s \to t$ exists for all right dense points $t \in T$ [1]. A function $f : T \to T$ is said to be differentiable at $t \in T^k = \{T \setminus (\rho(t) \max(T), \max t)\}$ if

$$\lim_{\sigma(t) \to s} \frac{f(\sigma(t) - f(s))}{\sigma(t) - s},$$

where $s \in T - \{\sigma(t)\}$ exists and is said to be differentiable on T provided it is differentiable for each $t \in T^k$. A function $F : T \to T$, with $F^{\Delta}(t) = f(t)$ for all $t \in T^K$ is said to be integrable, if

$$\int_s^t f(\tau)\,\Delta\tau = F(t) - F(s),$$

where F is the anti-derivative of f and for all $s, t \in T$. Let $f : T \to T$ and if $T = R$ and $a, b \in T$, then $f^{\Delta}(t) = f^{1}(t)$ and

$$\int_a^b f(t)\,dt = \int_a^b f(t)\,\Delta t.$$

Further, if $T = Z$ (discrete case), then $f^{\Delta}(t)=\Delta f(t)=f(t+1)-f(t)$ and

$$\int_a^b f(t)\,\Delta t=\begin{cases}\sum_{k=a}^{b-1} f(k) \text{ if } a<b,\\ 0 \text{ if } a=b,\\ \sum_{k=b}^{a-1} f(k) \text{ if } a>b.\end{cases}$$

If $f, g: T \to X$ (X is a Banach space) be differentiable in $t \in T^k$. Then for any two scalars α, β, the mapping $\alpha f+\beta g$ is differentiable in t and further we have:

1. $(\alpha f+\beta g)^{\Delta}(t)=\alpha f^{\Delta}(t)+\beta g^{\Delta}(t)$;
2. $(fg)^{\Delta}(t)=f^{\Delta}(t)\,g(t)+f(\sigma(t))g^{\Delta}(t)$;
3. $f(\sigma(t))=f(t)+\mu(t)f^{\Delta}(t)$;
4. $(kf)^{\Delta}(t)=kf^{\Delta}(t)$, for any scalar k.

Note that if f is Δ-differentiable, then f is continuous. Further if t is right scattered and f is continuous at t then

$$f^{\Delta}(t)=\frac{f(\sigma(t))-f(t)}{\mu(t)}.$$

For a survey on calculus of Δ-differentiable functions, we refer to Lakshmikantham et. al [5].

3. Controllability and observability criteria for Δ-differentiable functions. In this section, we shall be concerned with the first order Δ-differentiable dynamic system

$$x^{\Delta}(t)=A(t)x(t)+B(t)U(t),\ x(t_0)=x_0, \tag{1}$$

$$y(t)=C(t)x(t), \tag{2}$$

where $A(t)$ is an $(n\times n)$ square matrix and $A: T^k \to B(R^n)$ is regressive and rd-continuous. When $t = R$, (1) is equivalent to

$$x'(t)=A(t)x(t)+B(t)U(t), x(t_0)=x_0, \tag{3}$$

and when $t = z$, (1) is equivalent to

$$x(n+1)=A(n)x(n)+B(n)U(n),\ x(n_0)=x_0. \tag{4}$$

The fundamental concepts of controllability and observability for an m-input, p-output, n-dimentional linear state equation (1) and (2) will be considered in this section. For a time varying linear state equation (3), the connection of the input signal to the state variables can change with time. Therefore, the concept of controllability is tied to a specific finite time interval $[t_0, t_f]$ with, of course $t_f > t_0$. For a discrete system (4), the connection of the input signal to the next state variables can change with time.

Definition 1. The Δ-differentiable dynamic system (1) is said to be controllable on $[t_0, t_f]$, if for any given initial state $x(t_0) = x_0$ $(x(n_0) = x_0)$, there exists a continuous (discrete) input signal $U(t)$, such that the corresponding solution of (1) satisfies $x(t_f) = 0$ $(x(n_f) = 0)$.

If time scale dynamical system (1) is controllable for all x_0 at $t = t_0$ and for all x_f at $t = t_f$, then the system (1) is said to be completely controllable. We suppose that $T^k = (a, b) \cap T$ and the associated homogeneous system is

$$x^{\Delta}(t) = A(t)\, x(t),\; x(t_0) = x_0.$$

Let $\Phi_A(t, t_0)$ be a fundamental matrix solution of

$$x^{\Delta}(t) = A(t)\, x(t).$$

Then any solution $x(t)$ of (1) has the form

$$x(t) = \Phi_A(t, t_0)\, x_0 + \int_{t_0}^{t} \Phi_A(t, \sigma(s))\, B(s)\, U(s)\, \Delta(s),$$

and it is easy to see that

$$\bar{x}(t) = \int_{t_0}^{t} \Phi_A(t, \sigma(s))\, B(s)\, U(s)\, \Delta(s)$$

is a particular solution of the dynamic system (1) [1].

We are now in a position to develop criteria for the dynamic system (1) to be completely controllable and observable. We have the following theorem.

Theorem 1. The time scale dynamical system is completely controllable on the closed interval $J = [t_0, t_f]$ if and only if the $(n \times n)$ symmetric matrix

$$W(t_0, t_f) = \int_{t_0}^{t_f} \Phi(t_0, \sigma(s))\, B(s)\, B^*(s)\, \Phi^*(t, \sigma(s))\Delta(s)$$

is non-singular.

P r o o f. We first suppose that $W(t_0, t_f)$ is nonsingular. Then it is claimed that the dynamic system (1) is completely controllable. For given an $(n \times 1)$ vector x_0, choose

$$U(t) = -B^*(t)\, \Phi^*(t, \sigma(t))\, W^{-1}(t_0, t_f)\, x_0.$$

Clearly, the input signal U is continuous on J and the corresponding general solution of (1) with the initial condition $x(t_0)=x_0$ is given by

$$x(t_f)=\Phi(t_f,t_0)x_0+\int_{t_0}^{t_f}\Phi(t_0,\sigma(s))B(s)U(s)\Delta(s).$$

Substitute for $U(t)$ and using the definition of $W(t_0,t_f)$, we get

$$x(t_f)=\Phi(t_f,t_0)x_0-$$

$$-\int_{t_0}^{t_f}\Phi(t_f,\sigma(s))B(s)B^*(s)\Phi^*(t_0,\sigma(s))W^{-1}(t_0,t_f)x_0\Delta(s)=\Phi(t_f,t_0)x_0-$$

$$-\Phi(t_f,t_0)\int_{t_0}^{t_f}\Phi(t_0,\sigma(s))B(s)B^*(s)\Phi(t_0,\sigma(s))W^{-1}(t_0,t_f)x_0\Delta(s)=0.$$

Thus the dynamic system is controllable. This is true for all $t_0 \le t \le t_f$, it follows that the system (1) is completely controllable.

Next suppose that the dynamic system (1) is completely controllable on J and suppose that $W(t_0,t_f)$ is singular. Then since $W(t_0,t_f)$ is non-invertible there exists a non-zero $(n\times 1)$ vector y such that

$$y^*W(t_0,t_f)C=\int_{t_0}^{t_f}y^*\Phi(t_0,\sigma(s))B(s)B^*(s)\Phi^*(t_0,\sigma(s))y(s)\Delta(s).$$

Because of the fact that the integrand in this expression is non-negative continuous function, we have

$$\|y^*\Phi(t_0,\sigma(s))B(s)\|=0,$$

it follows that

$$y^*\Phi(t_0,\sigma(s))B(s)=0,\ s\in J. \tag{5}$$

Since the state equation is completely controllable on J, choose $x_0=y$, there exists a continuous input $U(t)$ such that

$$0=\Phi(t_f,t_0)y+\int_{t_0}^{t_f}\Phi(t_f,\sigma(s))B(s)U(s)\Delta(s)$$

or

$$y=-\int_{t_0}^{t_f}\Phi^{-1}(t_f,t_0)\Phi(t_f,\sigma(s))B(s)U(s)\Delta(s)=$$

$$= -\int_{t_0}^{t_f} \Phi(t_0, \sigma(s)) B(s) U(s) \Delta(s).$$

Thus,

$$y^* y = -\int_{t_0}^{t_f} y^* \Phi(t_0, \sigma(s)) B(s) U(s) \Delta(s),$$

and since (5) holds, it follows that $y^* y = 0$, and thus it contradicts the fact that $y \neq 0$. Thus $W(t_0, t_f)$ is non-singular and the proof of the theorem is complete.

We now develop an algorithm corresponding to the time scale dynamical system (1). We define a sequence of $(n \times m)$ matrix functions

$$K_0(t) = B(t), \tag{6}$$

$$K_j(t) = A(t) K_{j-1}(t) + K_{j-1}^{\Delta}(t), \; j = 1, 2, 3, \ldots . \tag{7}$$

We observe the following: $\Phi(t, \sigma(s)) = \Phi(t) \Phi^{-1}(\sigma(s))$, when $T = R$, $\sigma(s) = s$ and $\Phi(t, s) = \Phi(t) \Phi^{-1}(s)$. On the other hand when $T = Z$, $\sigma(s) = s + 1$ and $\Phi(t, s+1) = \Phi(t) \Phi^{-1}(s+1)$. Further

$$\Phi^{\Delta}(t, \sigma(s)) = A(t) \Phi(t, \sigma(s))$$

and

$$[\Phi(t, \sigma(s))^{\Delta}] = [\Phi(t) \Phi^{-1}(\sigma(s))]^{\Delta} =$$

$$= \Phi(t) [-\Phi^{-1}(\sigma(s)) A(\sigma(s))] = -\Phi(t) \Phi^{-1}(\sigma(s)) A(\sigma(s)).$$

Using the iteration idea given in (6), (7), we can easily verify the following:

$$[\Phi(t, \sigma(s)) B(\sigma(s))]^{\Delta^j} = \Phi(t, \sigma(s)) K_j(\sigma(s)), \; j = 0, 1, 2, \ldots,$$

where

$$K_j(\sigma(s)) = -A(\sigma(s)) K_{j-1}(\sigma(s)) + K_{j-1}^{\Delta}(\sigma(s)), \; j = 0, 1, 2, \ldots .$$

When $\sigma(s) = s = t$, we have

$$K_j(t) = [\Phi(t, s) B(s)]_{s=t}^{\Delta}, \; j = 0, 1, 2, \ldots .$$

Based on the above iterative criteria, we have the following theorem.

Theorem 2. Suppose m is a positive integer such that for all $t \in [t_0, t_f]$, B is m times continuously Δ-differentiable and A is $(m-1)$ times continuously Δ-differentiable. Then the linear dynamic system (1) is completely controllable on $[t_0, t_f]$ if for some $t_c \in [t_0, t_f]$,

$$\text{Rank}\, [K_0(t_c), K_i(t_c), \ldots, K_m(t_c)] = n . \tag{8}$$

P r o o f. Suppose rank condition (8) holds for some $t_c \in [t_0, t_f]$. Then it is claimed that the dynamic system (1) is completely controllable. To the contrary, suppose the dynamic system (1) is not completely controllable on $[t_0, t_f]$. Then by theorem 1, the Grammian matrix $W[t_0, t_f]$ is non-invertible, and hence there exists an $(n \times 1)$ vector y such that

$$y^*\Phi(t_0, t)B(t) = 0, \ t \in (t_0, t_f). \tag{9}$$

Let y_1 be a non-zero vector such that $y_1 = \Phi(t_0, t_c)r$. Then from (9), we have $y_1^*\Phi(t_c, t)B(t) = 0$, $t \in (t_0, t_f)$. In particular, at $t = t_c$, we have $y_1^* K_0(t_c) = 0$. Now, Δ-differentiation with respect to t yields

$$y_1^*\Phi(t_c, t)K_1(t_c) = 0, \ t \in (t_0, t_f).$$

This implies $y_1^* K_i(t_c) = 0$ continuing in this way, we get $y_1^* K_j(t_c) = 0$ for $j = 0$, 1, 2, …, m. Therefore,

$$y_1^* \, [K_0(t_c), K_i = (t_c) \ldots K_m(t_c)] = 0,$$

and this contradicts the fact that (8) holds. Thus the proof of the theorem is complete.

We now turn our attention to the concept of observability on a time scale dynamical system. It is simpler to consider the case of zero input, and this does not entail any loss of generability since the concept is not altered in the presence of a known input signal. Therefore, we consider the unforced dynamical system

$$\begin{gathered} x^\Delta(t) = A(t)x(t), \ x(t_0) = x_0, \\ y(t) = C(t)x(t). \end{gathered} \tag{10}$$

Definition 2. The time scale dynamical system (10) is said to be completely observable on $[t_0, t_f]$, if for any initial state $x(t_0) = x_0$, it is uniquely determined by the corresponding response $y(t)$ for all $t \in [t_0, t_f]$.

We now present a necessary and sufficient condition for the system (10) to be completely observable.

Theorem 3. The time varying time scale dynamical system (10) is completely observable on $[t_0, t_f]$ if and only if the $(n \times n)$ symmetric observability matrix

$$M[t_0, t_f] = \int_{t_0}^{t_f} \Phi^*(s, t_0)C^*(s)C(s)\Phi(s, t_0)\,\Delta(s)$$

is non-singular.

P r o o f. Suppose that $M[t_0, t_f]$ is nonsingular. Then the solution expression with $U(t)=0$ is given by $y(t)=C(t)\Phi(t,t_0)x_0$, or

$$\Phi^*(t,t_0)C^*(t)y(t)=\Phi^*(t,t_0)C^*(t)C(t)\Phi(t,t_0)x_0.$$

Hence

$$\int_{t_0}^{t_f}\Phi^*(s,t_0)C^*(s)y(s)\Delta(s)=$$

$$=\int_{t_0}^{t_f}\Phi^*(s,t)C^*(s)C(s)\Phi(s,t_0)x_0\Delta(s)=M(t_0,t_f)x_0.$$

Since M is non-singular, x_0 can be determined uniquely. Thus the dynamical system (7) is completely observable.

Conversely, suppose the dynamic system (10) is completely observable. Then it can be easily proved as in Theorem 1, that $M(t_0,t_f)$ is nonsingular.

4. Real-time database systems. In this section, we shall be concerned with real time database systems and in fact concurrency control is one of the main issues of real time data base systems. Many real world applications contain time constraints to data as well as access to data that has temporal validity. Telecommunication is an example of an application area, which has database requirements that require a real-time database or at least time-cognizant database. Most database requests are simple reads, with access to few and return to some value based on the content in the database. Our main concern in this section is, is there a distributed concurrence control method that is suitable for a real-time database system.

Traditional databases deal with persistent data. Transactions access this data while maintaining consistency. The goal of transaction and query processing in database is to get a good response time. On the other hand, real time systems can also deal with temporal data; i.e., data that becomes outdated after a certain amount of time. Due to the temporal character of the data and the response time requirements forced by the nature, tasks in real time system have time constraints. The main purpose of this section is that time goal of real time system is met with jump operators. These jump operators play a crucial role in updating data and ignoring the outdated data and in softening the required information and object information. Efficient concurrency control protocols are required in order for it to be possible to schedule real-time database transactions. This is achieved by jump operators to schedule real-time database transactions. We first define the notion of a «real time». Given $t \in R$, we write $\hat{\sigma}_i(t)$ to denote the value of σ_i at time t interpreting the clock as a counter. That is $\hat{\sigma}_i(t)=\mathrm{Sup}\{n/\sigma_i(n)\le t\}$. Here supremum is used in the maximum sense. If the supremum is taken over an empty set, then it is zero, that is $\hat{\sigma}_i(t) = 0$ if the clock is not ticked at all until time t. It may be noted that a real-time transaction is a transaction with additional real-time attributer: deadline, priority and importance. These attributes are used by scheduling the real time algorithm and concurrency control method [6, 7].

We assume that every site contains a directory containing all objects and their location. Further, every site contains data structures for keeping transaction and object information. The transaction data structure contains information of transaction and object information. The transaction data structure contains information of transaction identification, the phase where a transaction is, transaction's read and write sets, and other information like administration. Before a transaction can enter the read phase, we must first initialize data by using zero $(x) = 0$. Now the read phase starts with a begin operation. In the read phase if the transaction reads an object several checks must be done. We first note that a transaction requesting the data must be active and not aborted. Secondly, a requested data item must not be marked as an validating object. Finally, if the object is not located in the local node, a dead operation must be requested in the objects local node.

The importance of a real time database is its processing and its approach to resolve data and resource conflicts. In real-time databases, timely transaction execution is more important and both fairness and maximum resource utilization become secondary goals. Further, the real time databases use the percentage of transactions that complete within their deadlines. It is usually assumed that a hard transaction can never come into conflict with any other transaction and hard transactions cannot be aborted and will always complete successfully. Whereas soft transactions might be in conflict with other soft transaction, and, if two soft transactions attempt to obtain a read lock or write lock which violate the lock compatibility, then the results in late transactions are considered to be in conflict with each other. We first establish the following theorem which will be used for further discussion.

Theorem 4. A hard database transaction can never enter a state of deadlock caused by conflicts with any other database transaction.

P r o o f. Since hard database transaction can never be in conflict with any other transaction it follows that conflicts can occur only in soft transactions. These conflicts among soft transactions can be resolved in two ways : (1) If the conflicting transaction occurs at a lower priority than any other conflicting soft transaction, and has not entered the committing step, then it is aborted and thus resolving the conflict. (2) If the conflicting soft transaction is executing at a lower priority than any other conflicting soft transaction, and has entered the committing step will be blocked until the transaction is complete and thus releasing all its locks. Since a transaction, which has entered the committing step, can-

T	Event	Database State	Comments
T_1	INITIALIZE BOT	Zero (x) $\{x, y\}$	Zero T_1 starts
T_1	W-lock (x)	$W^{\Delta}(t) = \frac{DIFF\ W(\sigma(t), W(t))}{DIFF(t+1, t)}$	Lock is removed DIFF $(x, y) = x - y$
		$W(\sigma(t)) = DIFF\ (\sigma(t+1), \sigma(t))$	$W(\sigma(t)) = DIFF\ (\sigma(t+1), -\sigma(t))$
	SUCC (T_1)	T_2	Goes to the next event T_2
T_2	Write $(x) \to x'$	$\{x, y\}$	T_2 pre-empts T_1 and update x. T_2 is serialized after T_1.
T_3	Write $(y) \to y'$	$\{x', y'\}$	T_3 update y. Since y is not yet write locked on T_1. T_3 is serialized before T_1 according to Rule 2.
T_1	Upd $(y^1) \to y''$		T_1 update y however this update is visible for other transactions.
T_1	EOT	$\{x', y''\}$	T_1 ends and releases its lock. Y_{11} is now visible.

not obtain any further locks, it cannot cause any further conflicts with any other transaction, the proof is complete.

First, we initialize data and then local validation can be achieved by SUCCESOR [SUCC $(x) = x + 1$] function which acts like iterating all objects accessed by the transaction, finds conflicting operation (if any), and resolves conflicts. The adjustment of time stamp intervals iterates through the READ set and WRITE set of the validating transaction. This is achieved by the objects read and write time stamp. When access has been made to the same objects both in the validating transaction and in the active transaction, the temporal interval of the active transaction is adjusted by the jump operators. Thus we use deferred dynamic time adjustment of the serial order.

The following serialization rule applies to each transaction.

Rule 1. A set of executing soft transactions are serialized in the order they perform the end of transactions. This enables their changes visible for other transactions.

Rule 2. A hard transaction, reading or writing the value of a data element x, is serialized before all hard transactions reading or writing the value of x at a later time. Further, the transaction is serialized before any soft database transaction obtaining a lock on x at a later time.

Rule 3. A hard transaction, updating the value of a data element currently locked by a soft transaction, is serialized only after that transaction.

We first need to verify that whether or not transactions always read the correct version of a data element, i.e., the value produced by the last serialized transaction updating that particular transaction and no intermediate results produced by executing transactions are visible to other transactions i.e., we need to verify consistency of transactions.

Now we consider three transactions T_1, T_2 and T_3 executed in Table.

Note that T_1 is a soft transaction T_2 and T_3 are hard transactions. The algorithm can easily extended to n transactions $n \geq 3$. In order to avoid monotony we even omit formulating the algorithm.

From the example we see that the resulting serialization order is T_3, T_1 and T_2 even thorough the actual order of commit is T_2, T_3 and T_1. This serialization approach trades a relaxation of serialization for freshness of data.

Узагальнено результати досліджень керованості та спостережуваності при масштабуванні за часом. Отримано результати класичної теорії як окремого випадку, при цьому усунуто обмеження за часом при паралельному керуванні операторами переходу для динамічних систем, що масштабуються.

1. *Lakshmikantham V., Sivasundaram S., Kaymakelan B.* Dynamic Systems on Measure Chains. — Kluwer Academic Publishers, 1996.
2. *Aulbach B., Hilgeu S.* A Unified Approach to Continuous and Discrete Dynamics. Qualitative Theory of Differential Equations. — Szeged, Hungary 53. — 1983.
3. *Aulbach B., Hilgeu S.* Linear Dynamic Process with Inhomogeneous Time Scales. Nonlinear Dynamic and Dynamical Systems. — Berlin : Academic Verlag, 1980.
4. *Amer Abu Ali.* An Optimistic Concurrency Control for Real-time Database Systems// Amer. J. of Applied Sciences. — 2006. — N 3 (2). — P. 1706—1710.
5. *Barnett S., Cameron R. G.* Introduction to Mathematical Control Theory. — 2nd Edition. — Oxford University Press, 1985.
6. *Ramamritham K.* Real-Time Databases.// Int. J. of Distributed and Parallel Databases.— 1993. — **1**, N 2. — P. 199—226.
7. *Murty K. N., Rao Y. S.* Two Point Boundary Value Problems on Inhomogeneous Time Scale Dynamic Process // J. of Mathematical Analysis and Applications. — 1994. — **184**. — P. 22—34.

Submitted on 30.12.09