数理解析研究所講究録 1265 巻 2002 年 200-208

Performance Evaluations of Parallel RKN Methods

Nguyen Huu Cong

Faculty of Mathematics, Mechanics and Informatics
Hanoi University of Science
334 Nguyen Trai, Thanh Xuan, Hanoi, Vietnam

Abstract

This paper gives a survey on the development of parallel Runge-Kutta-Nyström (RKN) methods for the numerical solution of systems of special second-order ordinary differential equations (ODEs) of the form $\mathbf{y}'' = \mathbf{f}(t, \mathbf{y}(t))$. We shall consider two cases: *explicit parallel RKN methods* and *implicit parallel RKN methods*. In both cases, an implicit RKN method is used as a corrector method (RKN corrector) which is solved by an iteration scheme. In the first case, the iteration process is *explicit*. Whereas in the second case, the iteration process is *implicit*. The resulting iteration methods are well tuned to the parallel machines which reduce very much the sequential computational costs.

Key words: RKN methods, PC methods, parallelism.
AMS(MOS) subject classifications (1991): 65M12, 65M20
CR subject classifications: G.1.7

1 RKN methods

We will be concerned with the numerical solution of the initial-value problem (IVP) for the systems of special second-order, ordinary differential equations (ODEs)

$$\frac{d^2\mathbf{y}(t)}{dt^2} = \mathbf{f}(\mathbf{y}(t)), \quad \mathbf{y}(t_0) = \mathbf{y}_0, \quad \mathbf{y}'(t_0) = \mathbf{y}'_0, \qquad t_0 \le t \le T, \quad \mathbf{y}, \mathbf{f} \in \mathbb{R}^d \tag{1.1}$$

by Runge-Kutta-Nyström methods (RKN methods) on parallel computers. One (simple) option for solving problem (1.1) consists in writing the problem in first-order form and applying a parallel integration method for first-order ODEs, without taking into account the special form of this problem (the "indirect" approach). However, ignoring the fact that the right-hand side function $\mathbf{f}$ does not contain the first derivative, usually leads to algorithms that are less efficient than algorithms tuned to the special form of (1.1) (the "direct" approach). We illustrate this by an example from the class of sequential Runge-Kutta (RK) type methods. The highest-order, explicit RK method for first-order ODEs available in the literature, is the 17-stage, tenth-order RK method from [31]. Thus, writing (1.1) in first-order form and applying this RK method requires 17 sequential right-hand

side evaluations per integration step. Alternatively, we can pick a RKN method directly designed for problems like (1.1). In [33], we can find an RKN method of order 10 requiring 11 right-hand side evaluations per integration step. Hence, in this example, exploiting the special form of the differential equations, saves 6 right-hand side evaluations per step.

The mentioned example compares the direct and indirect approach for sequential methods. It is highly likely that in the class of parallel methods, the direct approach will also lead to an improvement of the efficiency. This motivated us to develop direct parallel RKN methods for solving problem (1.1), rather than using existing parallel methods for first-order problems via the indirect approach.

Thus our starting point is the s-stage RKN corrector method (RKN corrector)

$$\mathbf{Y}_{n+1} = \mathbf{e}\otimes\mathbf{u}_n + h\mathbf{c}\otimes\mathbf{u}'_n + h^2\mathbf{a}\otimes\mathbf{f}(\mathbf{u}_n) + h^2(A\otimes I)\mathbf{F}(\mathbf{Y}_{n+1}), \tag{1.2a}$$

$$\begin{aligned}\mathbf{u}_{n+1} &= \mathbf{u}_n + h\mathbf{u}'_n + h^2 b_0\mathbf{f}(\mathbf{u}_n) + h^2(\mathbf{b}^T\otimes I)\mathbf{F}(\mathbf{Y}_{n+1}),\\ \mathbf{u}'_{n+1} &= \mathbf{u}'_n + hd_0\mathbf{f}(\mathbf{y}_n) + h(\mathbf{d}^T\otimes I)\mathbf{F}(\mathbf{Y}_{n+1}).\end{aligned} \tag{1.2b}$$

Here, s-by-s matrix A, s-dimensional vectors $\mathbf{c}$, $\mathbf{a}$, $\mathbf{b}$, $\mathbf{d}$ and scalars b_0, d_0 are the method parameters, $\mathbf{e}$ is the s-dimensional vector with unit entries, I is the d-by-d identity matrix, $h = t_{n+1} - t_n$ is the stepsize, and $\otimes$ denotes the Kronecker product. The sd-dimensional vector $\mathbf{Y}_{n+1} = \left(\mathbf{Y}^T_{n+1,1},\ldots,\mathbf{Y}^T_{n+1,s}\right)^T$ is the *stage vector* representing the numerical approximations to the exact solution vector $\mathbf{y}(\mathbf{e}t_n + \mathbf{c}h) = [\mathbf{y}^T(t_n + c_1h),\ldots,\mathbf{y}^T(t_n + c_sh)]^T$, $\mathbf{F}(\mathbf{Y}_{n+1}) = \left(\mathbf{f}^T(\mathbf{Y}_{n+1,1}),\ldots,\mathbf{f}^T(\mathbf{Y}_{n+1,s})\right)^T$, $\mathbf{u}_n \approx \mathbf{y}(t_n)$, $\mathbf{u}_{n+1} \approx \mathbf{y}(t_{n+1})$, $\mathbf{u}'_n \approx \mathbf{y}'(t_n)$ and $\mathbf{u}'_{n+1} \approx \mathbf{y}'(t_{n+1})$. If the matrix A is strictly lower triangular, then the RKN method (1.2) is called *explicit*. Otherwise, it is called *implicit*. The sequential explicit RKN methods of orders up to 10 can be found in [24] - [30], [33]. For the sequential implicit RKN methods of arbitrary orders, we refer the readers to [32] and [35]. We see from these papers that there are two families of implicit RKN methods.

The methods of the first family are called *indirect* RKN methods. They are obtained by writing (1.1) in first-order form and applying the implicit RK defined by $\{\mathbf{c}, A_{rk}, \mathbf{b}_{rk}\}$. The resulting indirect RKN method is defined by $\{\mathbf{c}, A = [A_{rk}]^2, \mathbf{b} = A_{rk}\mathbf{b}_{rk}, \mathbf{d} = \mathbf{b}_{rk}\}$.

The methods of the second family are called *direct* RKN methods. They are directly constructed for second-order form (1.1). The indirect and direct RKN methods of collocation type based on the same collocation vector $\mathbf{c}$ have the same order of accuracy. With respect to stability properties, the indirect RKN methods are better than direct RKN methods (cf. [35]).

2 parallel RKN methods

For parallel numerical solution of IVP (1.1), we consider the following method

$$\mathbf{Y}^{(0)}_{n+1} = \mathbf{e}\otimes\mathbf{y}_n + h\mathbf{c}\otimes\mathbf{y}'_n + h^2\mathbf{a}\otimes\mathbf{f}(\mathbf{y}_n) + h^2(C\otimes I)\mathbf{F}(\mathbf{Y}^{(0)}_{n+1}), \tag{2.1a}$$

$$\begin{aligned}\mathbf{Y}^{(j)}_{n+1} &= \mathbf{e}\otimes\mathbf{y}_n + h\mathbf{c}\otimes\mathbf{y}'_n + h^2\mathbf{a}\otimes\mathbf{f}(\mathbf{y}_n) + h^2(B\otimes I)\mathbf{F}(\mathbf{Y}^{(j)}_{n+1})\\ &\quad + h^2\big((A-B)\otimes I\big)\mathbf{F}(\mathbf{Y}^{(j-1)}_{n+1}), \quad j = 1,\ldots,m,\end{aligned} \tag{2.1b}$$

$$\begin{aligned}\mathbf{y}_{n+1} &= \mathbf{y}_n + h\mathbf{y}'_n + h^2 b_0\mathbf{f}(\mathbf{y}_n) + h^2(\mathbf{b}^T\otimes I)\mathbf{F}(\mathbf{Y}^{(m)}_{n+1}),\\ \mathbf{y}'_{n+1} &= \mathbf{y}'_n + hd_0\mathbf{f}(\mathbf{y}_n) + h(\mathbf{d}^T\otimes I)\mathbf{F}(\mathbf{Y}^{(m)}_{n+1}),\end{aligned} \tag{2.1c}$$

where B and C are appropriately chosen matrices and m is an integer. In the case of stiff problems (see [6, pp. 4-5]), it is recommendable to replace (2.1c) by the formula

$$\begin{aligned} \mathbf{y}_{n+1} &= \mathbf{y}_n + h\mathbf{y}'_n + h^2 b_0 \mathbf{f}(\mathbf{y}_n) \\ &\quad + (\mathbf{b}^T A^{-1} \otimes I)(\mathbf{Y}_{n+1}^{(m)} - \mathbf{e} \otimes \mathbf{y}_n - h\mathbf{c} \otimes \mathbf{y}'_n - h^2 \mathbf{a} \otimes \mathbf{f}(\mathbf{y}_n)), \\ \mathbf{y}'_{n+1} &= \mathbf{y}'_n + h d_0 \mathbf{f}(\mathbf{y}_n) \\ &\quad + \frac{1}{h}(\mathbf{d}^T A^{-1} \otimes I)(\mathbf{Y}_{n+1}^{(m)} - \mathbf{e} \otimes \mathbf{y}_n - h\mathbf{c} \otimes \mathbf{y}'_n - h^2 \mathbf{a} \otimes \mathbf{f}(\mathbf{y}_n)), \end{aligned} \tag{2.1d}$$

provided that the matrix A is nonsingular. The method (2.1) can be interpreted as an iterative method with m iterations. Evidently, if $m \to \infty$ and if $\mathbf{Y}_{n+1}^{(m)}$ converges, then $\mathbf{Y}_{n+1}^{(m)}$ converges to the solution $\mathbf{Y}_{n+1}$ of (1.2a). However, for fixed m, we may also interpreted (2.1) as an RKN method. We call (2.1a) the *predictor formula* (predictor method or predictor), {(2.1b), (2.1c)} or {(2.1b), (2.1d)} the *corrector formula* (corrector step point formula).

The order of accuracy, the linear stability and the amount of intrinsic parallelism of the methods (2.1) are determined by the matrices A, B and C. We have the following general result for the (nonstiff) order of accuracy.

Theorem 2.1 *The order of accuracy of the RKN methods* {(2.1a), (2.1b), (2.1c)} *and* {(2.1a), (2.1b), (2.1d)} *are respectively given by* $p^* = \min\{p, 2m + q + 1\}$ *and* $p^* = \min\{p, 2m + q - 1\}$, *where* p *and* q *denote the order of the corrector method* (1.2) *and the order of predictor formula* (2.1a) *for* $\mathbf{Y}_{n+1}^{(0)}$.

From now on, we assume that the order of a method is always meant to be the *nonstiff* order of accuracy.

The linear stability properties can be obtained by applying (2.1) to the model test equation $y''(t) = \lambda y(t)$. This leads us to the recursion $(y_{n+1}, hy'_{n+1})^T = M_m(z)(y_n, hy'_n)^T$, $z = \lambda h$. For the step point formulas (2.1c) and (2.1d), the amplification matrices $M_m(z)$ are respectively defined as

$$M_m(z) = \begin{pmatrix} 1 + z[b_0 + \mathbf{b}^T P_m(z)(\mathbf{e} + z\mathbf{a})] & 1 + z\mathbf{b}^T P_m(z)\mathbf{c} \\ z[d_0 + \mathbf{d}^T P_m(z)(\mathbf{e} + z\mathbf{a})] & 1 + z\mathbf{d}^T P_m(z)\mathbf{c} \end{pmatrix} \quad \text{and} \tag{2.2a}$$

(2.2b)

$$M_m(z) = \begin{pmatrix} 1 + zb_0 - \mathbf{b}^T A^{-1}[I - P_m(z)](\mathbf{e} + z\mathbf{a}) & 1 - \mathbf{b}^T A^{-1}[I - P_m(z)]\mathbf{c} \\ zd_0 - \mathbf{d}^T A^{-1}[I - P_m(z)](\mathbf{e} + z\mathbf{a}) & 1 - \mathbf{d}^T A^{-1}[I - P_m(z)]\mathbf{c} \end{pmatrix},$$

where the s-by-s matrix $P_m(z)$ is given by

$$P_m(z) = [I - zB]^{-1}\{[I - z(A - B)]^{-1}[I - z^m(A - B)^m] + z^m(A - B)^m[I - zC]^{-1}\}.$$

The spectral radius $\rho(M_m(z)$ of the amplification $M_m(z)$ is called the *stability function* of the parallel RKN methods.

In the following sections, we discuss the cases: (i) Both matrices B and C vanish; (ii) The matrix C vanishes and the matrix B is diagonal or Both matrices C and B are

2.1 Parallel explicit RKN methods

Let us assume that the problem (1.1) is stable and nonstiff, that is, the eigenvalues of the Jacobian matrix $\partial \mathbf{f}/\partial \mathbf{y}$ are assumed to be on the *negative* axis and not "too far away" from the origin. For this problem, we may choose the RKN corrector with $\mathbf{a} = \mathbf{0}$, step point formula (2.1c) and set in (2.1), $B = C = O$ (case (i): B and C vanish) to obtain an explicit $s(m+1)$-stage RKN method defined by

$$\mathbf{Y}_{n+1}^{(0)} = \mathbf{e} \otimes \mathbf{y}_n + h\mathbf{c} \otimes \mathbf{y}'_n, \tag{2.3a}$$

$$\mathbf{Y}_{n+1}^{(j)} = \mathbf{e} \otimes \mathbf{y}_n + h\mathbf{c} \otimes \mathbf{y}'_n + h^2(A \otimes I)\mathbf{F}(\mathbf{Y}_{n+1}^{(j-1)}) \quad j = 1, \ldots, m, \tag{2.3b}$$

$$\begin{aligned} \mathbf{y}_{n+1} &= \mathbf{y}_n + h\mathbf{y}'_n + h^2(\mathbf{b}^T \otimes I)\mathbf{F}(\mathbf{Y}_{n+1}^{(m)}), \\ \mathbf{y}'_{n+1} &= \mathbf{y}'_n + h(\mathbf{d}^T \otimes I)\mathbf{F}(\mathbf{Y}_{n+1}^{(m)}). \end{aligned} \tag{2.3c}$$

The method (2.3) requires $sm + s$ right-hand side evaluations per step. We see that each block of s stages of this method can be computed in parallel, so that on parallel computers with s processors, the number of sequential right-hand side evaluations (in each processor) will be only $s^* = m + 1$. Such methods are called in [5, 40] *parallel-iterated RKN method* or briefly *PIRKN method.*

Since in this case, $\mathbf{Y}_{n+1} - \mathbf{Y}_{n+1}^{(0)} = O(h^2)$, Theorem 2.1 implies that the order of accuracy of the PIRKN method is given by $p^* = \min\{p, 2m + 2\}$. The "optimal" resulting PIRKN method of order $p^* = p$ is obtained by setting $m = [(p-1)/2]$ requires $s^* = m + 1 = [(p+1)/2]$ sequential right-hand side evaluations per step. Consequently a tenth-order PIRKN method requires only $s^* = 5$ sequential right-hand side evaluations per step. Whereas tenth-order explicit RKN method of Hairer in [33] requires $s^* = 11$ sequential right-hand side evaluations per step (6 evaluations per step are saved). We note that the number of sequential right-hand side evaluations per step s^* of the similar tenth-order optimal PIRK method for first-order ODEs proposed in [36] is still equal to 10.

For the stability of the PIRKN methods, the amplification matrix $M_m(z)$ and the stability function $\rho(M_m(z))$ are defined with the matrix $P_m(z)$ in (2.2a) given by

$$P_m(z) = [I - zA]^{-1}[I - (zA)^{m+1}]$$

If we regard the PIRKN methods as real iteration ones, then the number of iterations m is determined by a *dynamic* iteration strategy and can be varied from step to step. The PIRKN methods in this situation are remained explicit RKN methods within a step. In this case, some improvements can be achieved by optimizing the rate of convergence and the size of iteration errors. This can be done by using direct RKN corrector methods (see [5]). A further improved efficiency can be obtained by considering parallel explicit RKN-type methods. This will be discussed in Section 3.

2.2 Parallel implicit RKN methods

Implicit RKN methods are applied to problems originating from structural mechanics or celestial mechanics, whose solutions possess periodic components with frequencies ranging from small to large, where the lower harmonics are of interest, the higher harmonics are not. Hence, only the solution components corresponding to eigenvalues of

the Jacobian matrix $\partial \mathbf{f}/\partial \mathbf{y}$ close to the origin are of interest. In such cases, the ideal method would be a method without dissipation of the lower harmonics (i.e., nonempty periodicity interval), high order of dispersion, and damping of the higher harmonics. The prevence of unwanted high harmonics (a form of stiffness) may reduce the step point order considerably. In many stiff problems, it is the stage order that determines the accuracy, rather than the step point order. In order to avoid the effect of order reduction, we need methods that have, in addition to a high step point order and the property of unconditional stability, a high stag order. In [39], the property of unconditional stability is termed R-stability. In our research papers (see e.g., [1, 2, 4, 35]), we have called it A-stability, in analogy with the terminology used for unconditionally stable methods for first-order ODEs.

For solving stiff problems, the step point formula (2.1d) should be used. Parallel $s(m+1)$-stage diagonally implicit RKN methods arise if in (2.1), the matrices C and B are diagonal or $C = O$, B is diagonal (case (ii)). Because of the diagonal implicitness, each block of s implicit stages can be computed in parallel, so that effectively, we only have $s^* = m+1$ implicit stages. If $C = O$, then the number of implicit stages is reduced to $s^* = m$. We call such methods *parallel diagonally implicit RKN methods* or briefly *PDIRKN methods*. Since for arbitrary matrix C, $\mathbf{Y}_{n+1} - \mathbf{Y}_{n+1}^{(0)} = O(h^2)$, Theorem 2.1 ensures that the order of accuracy of the PDIRKN method defined by {(2.1a), (2.1b), (2.1d)} is given by $p^* = \min\{p, 2m\}$. The best resulting PDIRKN methods of order $p^* = p$ have $s^* = [(p+1)/2]$ implicit stages.

In order to save computational costs involved with the LU-decomposition in solving implicit relations, we choose $B = C = D = \mathrm{diag}(\delta_1, \ldots, \delta_s)$ or $B = D = \mathrm{diag}(\delta_1, \ldots, \delta_s)$, $C = O$, and obtain the PDIRKN method of the form

$$\mathbf{Y}_{n+1}^{(0)} = \mathbf{e} \otimes \mathbf{y}_n + h\mathbf{c} \otimes \mathbf{y}'_n + h^2 \mathbf{a} \otimes \mathbf{f}(\mathbf{y}_n) + h^2(\theta D \otimes I)\mathbf{F}(\mathbf{Y}_{n+1}^{(0)}), \tag{2.4a}$$

$$\begin{aligned}\mathbf{Y}_{n+1}^{(j)} &= \mathbf{e} \otimes \mathbf{y}_n + h\mathbf{c} \otimes \mathbf{y}'_n + h^2 \mathbf{a} \otimes \mathbf{f}(\mathbf{y}_n) + h^2(D \otimes I)\mathbf{F}(\mathbf{Y}_{n+1}^{(j)}) \\ &\quad + h^2\big((A-D) \otimes I\big)\mathbf{F}(\mathbf{Y}_{n+1}^{(j-1)}), \quad j = 1, \ldots, m,\end{aligned} \tag{2.4b}$$

$$\begin{aligned}\mathbf{y}_{n+1} &= \mathbf{y}_n + h\mathbf{y}'_n + h^2 b_0 \mathbf{f}(\mathbf{y}_n) \\ &\quad + (\mathbf{b}^T A^{-1} \otimes I)(\mathbf{Y}_{n+1}^{(m)} - \mathbf{e} \otimes \mathbf{y}_n - h\mathbf{c} \otimes \mathbf{y}'_n - h^2 \mathbf{a} \otimes \mathbf{f}(\mathbf{y}_n)), \\ \mathbf{y}'_{n+1} &= \mathbf{y}'_n + h d_0 \mathbf{f}(\mathbf{y}_n) \\ &\quad + \frac{1}{h}(\mathbf{d}^T A^{-1} \otimes I)(\mathbf{Y}_{n+1}^{(m)} - \mathbf{e} \otimes \mathbf{y}_n - h\mathbf{c} \otimes \mathbf{y}'_n h^2 \mathbf{a} \otimes \mathbf{f}(\mathbf{y}_n)),\end{aligned} \tag{2.4c}$$

with $\theta \in \{0, 1\}$.

For the stability of the PDIRKN methods, the amplification matrix $M_m(z)$ and the stability function $\rho(M_m(z))$ for the case (ii) are defined with the matrix $P_m(z)$ in (2.2b) given by

$$P_m(z) = [I - zD]^{-1}\big\{[I - z(A-D)]^{-1}[I - z^m(A-D)^m] + z^m(A-D)^m[I - z\theta D]^{-1}\big\}$$

There are various strategies for choosing the positive entries of the diagonal matrices D (iteration parameters). One possibility is based on the minimization of the spectral radius of the stage vector iteration matrix (see [1, 7, 38]). For a large number of RKN correctors taken from the literature, we calculated the iteration parameters with this minimizing property. From these correctors, we selected those which generate methods

that remain A-stable after a minimal number of implicit sequential iterations. Let s^*, p^* and r^* denote this minimal number of implicit stages, the step point order, and the stage order, respectively. Then by means of the *minimal-spectral-radius* strategy, we found methods with $(s^*, p^*, r^*) = (3, 3, 2)$, $(5, 3, 4)$ and $(5, 5, 7)$. By replacing the condition "*the method should remain stable after a minimal number of iterations*" with the condition "*the method needs only to be A-stable if m is such that the order of the PDIRKN method equals that of the corrector*" (see [2, 4]), we found A-stable methods with $(s^*, p^*, r^*) = (4, 2, 2)$, $(6, 3, 3)$ and $(8, 4, 4)$. In order to appreciate these results, we mention the sequential RKN methods $(s^*, p^*, r^*) = (3, 1, 2)$ of Crouzeix in [22], $(s^*, p^*, r^*) = (4, 1, 3)$ of Sharp-Fine-Burrage in [39], and $(s^*, p^*, r^*) = (5, 1, 5)$ and $(6, 1, 5)$ of Cooper-Sayfy in [21].

3 Improving parallel RKN methods

In this section, we discuss some further ideas for improving the efficiency of parallel RKN methods. Because of complexity of the RKN methods for stiff problems and the limited length of the paper, we restricted our discussion to nonstiff problems.

The first idea is to replace the lower-order one-step predictor (2.3a) of the PIRKN method (2.3) with higher-order two-step predictors using past step point and stage values. Two following types of these two-step predictors have been considered (see g.e., [3, 10, 16])

$$\mathbf{Y}^0_{n+1} = \mathbf{w} \otimes \mathbf{y}_n + (V \otimes I)\mathbf{Y}^m_n : \qquad \text{Lagrange-type predictor}$$
$$\mathbf{Y}^0_{n+1} = \mathbf{e} \otimes \mathbf{y}_n + hc \otimes \mathbf{y}'_n + (B \otimes I)\mathbf{F}(\mathbf{Y}^m_n) : \qquad \text{Adams-type predictor}$$

The second idea is to design the new RKN-type corrector methods which optimize the convergence in the iteration process by means of minimizing spectral radius of iteration matrices, or new cheap RKN-type methods. A number of new methods have been constructed in the last few years like:

- Symmetric RKN methods (see [10, 16]).
- Two-step RKN methods (see [8, 11]).
- Pseudo two-step RKN methods (see [13, 17]).
- Explicit pseudo two-step RKN methods (see [14, 18, 19]).

The third idea is to increase the amount of parallelism in step-by-step methods by computing parallel solution values not only at step points, but also at off-step points. Thus, in each step, a whole block of approximations to the exact solution is computed. This approach was successfully used in [23] for obtaining reliable defect control in explicit RK methods. Alternatively, this approach can be used for reducing the number of iterations in the iteration process. For example, the block of approximations can be used for obtaining a very high-order predictor formula in the next step by some interpolation formulas e.g., Lagrange or Hermite interpolation (see [12, 20]). The high-order predictor can be obtained by the formula of Adams-type (see [15]). By choosing the abscissas of the off-step points narrowly spaced, we achieve much more accurate predictor values than can be obtained by two-step predictor formulas based on the past step point and stage values. Moreover, the precise location of the off-step points can be used for minimizing the iteration errors or for maximizing stability regions.

4 Concluding remarks

In this paper we have described some main ideas for construction and analysis of parallel RKN methods. The resulting new investigated methods are shown to be promising integration methods for the numerical solution of special second-order ODEs on parallel computers. Numerical experiments showed the superiority of these Parallel RKN methods over the best extant sequential methods in the RK and RKN literatures.

References

[1] N.H. Cong, *Parallel implicit Runge-Kutta-Nyström methods of direct collocation type for stiff initial-value problems*, J. Math. **20** (1992), 48-62.

[2] N.H. Cong, *Parallel direct collocation-based implicit Runge-Kutta- Nyström methods with high stability*, Acta Math. Viet. **17** (1992), 149-161.

[3] N.H. Cong, *An improvement for parallel-iterated Runge-Kutta-Nyström methods*, Acta Math. Viet. **18** (1993), 295-308.

[4] N.H. Cong, *A-stable diagonally implicit Runge-Kutta-Nyström methods for parallel computers*, Numerical Algorithms **4** (1993), 263-281.

[5] N.H. Cong, *Note on the performance of direct and indirect Runge-Kutta- Nyström methods*, J. Comput. Appl. Math. **45** (1993), 347-355.

[6] N.H. Cong, *Parallel Runge-Kutta-Nyström Methods*, PhD Thesis, University of Amsterdam (1994).

[7] N.H. Cong, *A fast convergence parallel DIRKN method and its applications to PDEs*, Appl. Math. Lett. **8** (1994), 85-90.

[8] N.H. Cong, *Direct collocation-based two-step Runge-Kutta-Nyström methods*, SEA Bull. Math. **19** (1995), 49-58.

[9] N.H. Cong, *Parallel Runge-Kutta and Runge-Kutta-Nyström Type Methods*, Dr.Sc. Thesis, Vietnam National University, Hanoi (1995).

[10] N.H. Cong, *Explicit symmetric Runge-Kutta-Nyström methods for parallel computers*, Computers Math. Applic. **31** (1996), 111-122.

[11] N.H. Cong, *Explicit parallel two-step Runge-Kutta-Nyström methods*, Computers Math. Applic. **32** (1996), 119-130.

[12] N.H. Cong, *RKN-type parallel block predictor-corrector methods with Lagrange-type predictors*, Computers Math. Applic. **35** (1998), 45-57.

[13] N.H. Cong, *Half-implicit pseudo two-step RKN methods*, SEA Bull. Math. **23** (1999), 585-597.

[14] N.H. Cong, *Explicit pseudo two-step RKN methods with stepsize control*, Appl. Numer. Math. **38** (2001), 135-144.

[15] N.H. Cong and N.T.H. Minh, *Parallel block PC methods with RKN-type correctors and Adams-type predictors*, Intern. J. Computer Math. **74** (2000), 509-527.

[16] N.H. Cong and N.T.H. Minh, *Fast convergence PIRKN-type PC methods with Adams-type predictors*, Intern. J. Computer Math. **77** (2001), 373-387.

[17] N.H. Cong and N.T.H. Minh, *Parallel-iterated pseudo two-step Runge-Kutta-Nyström methods for nonstiff second order IVPs*, Computers Math. Applic. **44** (2002), 143-155.

[18] N.H. Cong, H. Podhaisky and R. Weiner, *Performance of explicit pseudo two-step RKN methods on a shared memory computer*, Report No. **21** (2000), Inst. Numer. Math., Univ. Halle, Germany, http://www.mathematik.uni-halle.de/reports

[19] N.H. Cong, K. Strehmel and R. Weiner, *A general class of explicit pseudo two-step RKN methods on parallel computers*, Computers Math. Applic. **38** (1999), 17-30.

[20] N.H. Cong, K. Strehmel, R. Weiner and H. Podhaisky, *Runge-Kutta-Nyström-type parallel block predictor-corrector methods*, Advances in Computational Mathematics **10** (1999), 115-133.

[21] G.J. Cooper and A. Sayfy, *Semi-explicit A-stable Runge-Kutta methods*, Math. Comp. **33** (1979), 541-556.

[22] M. Crouzeix, *Sur l'approximation des équations différentielles opérationnelles linéaires par des méthodes de Runge-Kutta*, Ph.D Thesis, Université de Paris, France (1975).

[23] W.H. Enright and D.J. Highman, *Parallel defect control* BIT **31**, 647-663.

[24] E. Fehlberg, *Klassische Runge-Kutta-Nyström Formeln mit Schrittweiten-Kontrolle für Differentialgleichungen* $x'' = f(t,x)$, Computing **10** (1972), 305-315.

[25] E. Fehlberg, *Eine Runge-Kutta-Nyström Formel 9-ter Ordnung mit Schrittweitenkontrolle für Differentialgleichungen* $x'' = f(t,x)$, Z. Angew. Math. Mech. **61** (1981), 477-485.

[26] E. Fehlberg, S. Filippi und J. Gräf, *Eine Runge-Kutta-Nyström Formelpaar der Ordnung 10(11) für Differentialgleichungen* $y'' = f(t,y)$, Z. Angew. Math. Mech. **66** (1986), 265-270.

[27] S. Filippi und J. Gräf, *Ein Runge-Kutta-Nyström Formelpaar der Ordnung 11(12) für Differentialgleichungen der Form* $y'' = f(t,y)$, Computing **34** (1985), 271-282.

[28] S. Filippi and J. Gräf, *New Runge-Kutta-Nyström formula-pairs of order 8(7), 9(8), 10(9) and 11(10) for differential equations of the form* $y'' = f(t,y)$, J. Comput. Appl. Math. **14** (1986), 361-370.

[29] J.M. Fine, *Low order Runge-Kutta-Nyström methods*, Computing **38** (1987), 281-297.

[30] E. Hairer, *Méthodes de Nyström pour l'équation differentielle* $y''(t) = f(t,y)$, Numer. Math. **27** (1977), 283-300.

[31] E. Hairer, *A Runge-Kutta method of order 10*, J. Inst. Math. Appl. **21** (1978), 47-59.

[32] E. Hairer, *Unconditionally stable methods for second order differential equations*, Numer. Math. **32** (1979), 373-379.

[33] E. Hairer, *A one-step method of order 10 for* $y''(t) = f(t,y)$, IMA J. Numer. Anal. **2** (1982), 83-94.

[34] E. Hairer, S.P. Nøsett and G. Wanner, *Solving Ordinary Differential Equations, I. Nonstiff Problems*, 2nd revised edition (Springer-Verlag, Berlin, 1993).

[35] P.J. van der Houwen, B.P. Sommeijer and N.H. Cong, *Stability of collocation-based Runge-Kutta-Nyström methods*, BIT **31** (1991), 469-481.

[36] P.J. van der Houwen and B.P. Sommeijer, *Parallel iteration of high-order Runge-Kutta methods with stepsize control*, J. Comput. Appl. Math. **29** (1990), 111-127.

[37] P.J. van der Houwen, B.P. Sommeijer and W. Couzy, *Embedded diagonally implicit Runge-Kutta algorithms on parallel computers*, Math. Comp. **58** (1992), 135-159.

[38] P.J. van der Houwen, B.P. Sommeijer and N.H. Cong, *Parallel diagonally implicit Runge-Kutta-Nyström methods*, Appl. Numer. Math. **9** (1992), 111-131.

[39] P.W. Sharp, J.H. Fine and K. Burrage, *Two-stage and three-stage diagonally implicit Runge-Kutta-Nyström methods of orders three and four*, IMA J. Numer. Anal. **10** (1990), 489-504.

[40] B.P. Sommeijer, *Explicit, high-order Runge-Kutta-Nyström methods for parallel computers*, Appl. Numer. Math. **13** (1993), 221-240.