Particle Model to Optimize Enterprise Computing

Dianxun Shuai\textsuperscript{1}, Qing Shuai\textsuperscript{2}, Yuzhe Liu\textsuperscript{1}, and Liangjun Huang\textsuperscript{1}

\textsuperscript{1} East China University of Science and Technology, Shanghai 200237, P.R. China
shdx411022@online.sh.cn
\textsuperscript{2} Huazhong University of Science and Technology, Wuhan 430074, P.R. China
echoshuai@163.com

\textbf{Abstract.} This paper presents a novel generalized particle model (GPM) for the parallel optimization of enterprise computing\textsuperscript{*}. Since enterprise computing always involves the resource allocation, task assignment, and behavior coordination, without loss of generality, the proposed GPM is devoted to the optimization of enterprise computing in the context of the resource allocation and task assignment in complex environment. GPM transforms the optimization of enterprise computing into the kinematics and dynamics of massive particles in a force-field. The GPM approach has many advantages in terms of the high-scale parallelism, multi-objective optimization, multi-type coordination, multi-degree personality, and the ability to handle complex factors. Simulations have shown the effectiveness and suitability of the proposed GPM approach to optimize the enterprise computing.

\section{Introduction}

The distributed enterprise computing is featured by the geographically distributed resources and jobs, heterogeneous collection of autonomous systems, and collaboration based large-scale problem-solving. Since enterprise computing always involves the resource allocation, task assignment, and behavior coordination, their optimization in complex environment is of great significance for the quality-assurance and performance-improvement of enterprise computing.

\textsuperscript{*} This work was supported by the National Natural Science Foundation of China under Grant No. 60473044, No. 60575040 and No. 60135010.

---

\textit{Please use the following format when citing this chapter:}

There have been numerous algorithms and strategies for the optimization of enterprise computing. But most of them have the following limitations and disadvantages:

- Not suitable for enterprise computing in complex environment that may be related to multi-type coordinate, multi-degree autonomy, multi-objective optimization, multi-granularity coalition, and randomly and concurrently occurring phenomena.

- Just simpler coordination, such as cooperation and competition, is considered. Unilateral, unaware and unconscious co-ordinations are almost not taken into account.

- It is usually assumed that every entity of enterprise computing tries to increase either the aggregate utility of whole systems or its own personal utility. It turns out that all the entities seem to be either completely unselfish or completely selfish. Different autonomy and personality are not well embodied.

- The global control, global information exchange, and global objective are always required, so that enterprise computing can be realized only in series or small-scale parallel manner.

- The influence of the time-varying availability of entities, such as congestion, failure and priority level, on enterprise computing is not well considered. Particularly, the stochastic, emergent and concurrent nature of many phenomena can not be treated very well.

To overcome the above-mentioned limitations, this paper proposes a novel generalized particle model (GPM), which transforms the optimization of enterprise computing into the kinematics and dynamics of massive particles in a force-field. The features of the GPM-based enterprise computing optimization include the large-scale parallelism, multi-objective optimization, multi-type co-ordination, multi-degree autonomy, multi-granularity coalition, and the ability to deal with complex time-varying factors, e.g. the congestion, failure, and priority. Simulations have shown the effectiveness and suitability of the proposed GPM approach to the enterprise computing optimization.

2 Generalized Particle Model for Enterprise Computing

In order to formalize the enterprise computing optimization, consider the parallel distributed resource allocation among users. \( G(\tau) = \{G_1, \cdots, G_m\} \) be a finite set of resource users, and \( A(t) = \{A_1, \cdots, A_n\} \) be a finite set of resource suppliers in the time session \( \tau \). The supplier \( A_i \) provides the user \( G_j \) with the resource \( a_{ij}(t) \) at time \( t \), and meanwhile the user \( G_i \) offers the payment \( p_{ij}(t) \) for the unit resource of \( A_i \). The supplier \( A_i \) has the intention strength \( \zeta_{ij}(t) \) for the user \( G_j \), which is used to describe the effect of complex phenomena, such as interaction, congestion, failure, and priority. We thus obtain an assignment matrix \( S(t) = [s_{ik}(t)]_{n \times m} \) as shown in Fig. 1, where \( s_{ik}(t) = (a_{ik}(t), p_{ik}(t), \zeta_{ij}(t)) \). For convenience, they are normalized such that \( 0 \leq a_{ik}(t) \leq 1, 0 \leq p_{ik}(t) \leq 1 \) and \( -1 \leq \zeta_{ij} \leq +1 \).

The conceptual diagram of a generalized particle model (GPM) for the enterprise computing optimization is shown in Fig. 2, where the particle \( s_{ik} \) in a
force-field corresponds to the entry $s_{ik}$ in the assignment matrix $S$. A particle may be driven by several kinds of forces that are produced by the force-field, other particles and itself. The gravitational force produced by the force-field tries to drive a particle to move towards boundaries of the force-field, which embodies the tendency that a particle pursues maximizing the aggregate benefit of systems. The pushing or pulling forces produced by other particles are used to embody social coordination among resource suppliers and users. The self-driving force produced by a particle itself represents autonomy and personality of individual supplier and user of resources. Under the exertion of resultant forces, all the particles may move concurrently in a force-field. In this way, the GPM transforms the optimization problem of resource allocation for enterprise computing into the kinematics and dynamics of massive particles in a force-field.

$$
\begin{array}{c|c|c}
G_1 & \cdots & G_m \\
A_1 & a_{11}(t), p_{11}(t), \zeta_{11}(t) & a_{1m}(t), p_{1m}(t), \zeta_{1m}(t) \\
\vdots & \cdots & \cdots \\
A_n & a_{n1}(t), p_{n1}(t), \zeta_{n1}(t) & a_{nm}(t), p_{nm}(t), \zeta_{nm}(t) \\
\end{array}
$$

**Fig. 1.** The assignment matrix for the enterprise computing optimization, $S(t) = [s_{ik(t)}]_{k=1}^\infty$, where $s_{ik(t)} = (a_{ik}(t), p_{ik}(t), \zeta_{ik}(t))$

**Fig. 2.** Generalized particle model for the enterprise computing optimization.

**Definition 1.** Let $u_{ik}(t)$ be the utility of particle $s_{ik}$ at time $t$, and let $J(t)$ be the aggregate utility of all particles. They are defined by

$$
\begin{align*}
u_{ik}(t) &= a_{ik} \left[1 - \exp(-p_{ik}(t)a_{ik}(t))\right] \\
J(t) &= a \sum_{i=1}^n \sum_{k=1}^m u_{ik}(t)
\end{align*}
$$
where \( a_{ik}, a \geq 0 \), and \( a_{ik} \) is related to the activities of supplies \( A_i \) and user \( G_j \), such as congestion degree, failure rate, and priority level.

**Definition 2.** At time \( t \), the potential energy function \( P(t) \) that is related to the gravitational force of force-field \( F \) is defined by

\[
P(t) = \varepsilon^2 \ln \sum_{i=1}^{m} \sum_{k=1}^{m} \exp[-u_{ik}^2(t)/2 \varepsilon^2] - \varepsilon^2 \ln mn
\]

where \( 0 < \varepsilon < 1 \).

**Definition 3.** At time \( t \), the potential energy function \( Q(t) \) that is related to interactive forces among particles is defined by

\[
Q(t) = \xi \sum_{i=1}^{m} \left| \sum_{k=1}^{m} a_{ik} - r_i(t) \right|^2 - \sum_{i,j,k} \int_{0}^{\infty} \left[ (1 + \exp(-\zeta_{ik}x)) \right]^{-1} - 0.5 \, dx
\]

where \( 0 < \xi < 1 \); \( r_i \) is the capacity of resource supplier \( A_i \). The second term of \( Q(t) \) represents social co-ordinations among them, where \(-1 \leq \zeta_{ij} \leq +1\).

**Definition 4.** The hybrid energy function of the particle \( s_{ik} \) at time \( t \) is defined by

\[
\Gamma_{ik}(t) = -\lambda_{ik}^{(1)} u_{ik}(t) - \lambda_{ik}^{(2)} J_{ik}(t) + \lambda_{ik}^{(3)} P_{ik}(t) + \lambda_{ik}^{(4)} Q_{ik}(t)
\]

where \( 0 < \lambda_{ik}^{(1)}, \lambda_{ik}^{(2)}, \lambda_{ik}^{(3)}, \lambda_{ik}^{(4)} \leq 1 \).

**Definition 5.** Let the coordinate origin be located at the central line between the upper and bottom boundaries of force-field \( F \), and \( q_{ik}(t) \) be the vertical coordinate of particle \( s_{ik} \) at time \( t \). The dynamic equation for particle \( s_{ik} \) is defined by

\[
\begin{align*}
dq_{ik}(t)/dt &= \Psi_{ik}^{(1)}(t) - \Psi_{ik}^{(2)}(t) \\
\Psi_{ik}^{(1)}(t) &= -q_{ik}(t) + \gamma v_{ik}(t) \tag{6a} \\
\Psi_{ik}^{(2)}(t) &= I_{ik} + \sum_{j=1}^{m} w_{ik} u_{jk}(t) + \sum_{j=1}^{m} w_{ik} u_{jk}(t) \tag{6b}
\end{align*}
\]

where \( \gamma > 1 \), \( I_{ik} \) is a constant bias. The weight \( w_{ik} \) represents the polymerization strength of particles, \( s_{ik} \) and \( s_{ik} \), and \( w_{ij} \) represents the polymerization strength of particles, \( s_{ik} \) and \( s_{ij} \). The dynamic state \( v_{ik}(t) \) is a piecewise linear function of \( q_{ik}(t) \), which is defined by

\[
\begin{align*}
0 & \quad \text{if} \quad q_{ik}(t) < 0 \\
q_{ik}(t) & \quad \text{if} \quad 0 \leq q_{ik}(t) \leq 1 \\
1 & \quad \text{if} \quad q_{ik}(t) > 1 \tag{7}
\end{align*}
\]
Parallel Algorithm (GPMA):
Costep 1. Initiate in parallel $a_{ik}(t_0), p_{ik}(t_0), q_{ik}(t_0)$ for $i \in \{1, \cdots, n\}, k \in \{1, \cdots, m\}$.
Costep 2. By Eq.(1), calculate in parallel the utility $u_{ik}(t)$ of every particle $s_{ik}$ in force-field $F$ at time $t$;
Costep 3. Calculate in parallel $\Psi^{(1)}_{ik}(t)$ by Eq.(6a), and $\Psi^{(2)}_{ik}(t)$ by Eq.(6b) of every particle $s_{ik}$;
Costep 4. If all particles reach their equilibrium states at time $t$, then finish with success; Otherwise, modify $a_{ik}$ and $p_{ik}$ by the following Eqs.(8) and (9), respectively, then go to Costep 2.

$$\frac{dp_{ik}(t)}{dt} = \lambda^{(1)}_{ik} \frac{\partial u_{ik}(t)}{\partial p_{ik}(t)} + \lambda^{(2)}_{ik} \frac{\partial J(t)}{\partial p_{ik}(t)} - \lambda^{(3)}_{ik} \frac{\partial P(t)}{\partial p_{ik}(t)} - \lambda^{(4)}_{ik} \frac{\partial Q(t)}{\partial p_{ik}(t)} + \lambda^{(5)}_{ik} q_{ik}(t)$$  (8)

$$\frac{da_{ik}(t)}{dt} = \lambda^{(1)}_{ik} \frac{\partial u_{ik}(t)}{\partial a_{ik}(t)} + \lambda^{(2)}_{ik} \frac{\partial J(t)}{\partial a_{ik}(t)} - \lambda^{(3)}_{ik} \frac{\partial P(t)}{\partial a_{ik}(t)} - \lambda^{(4)}_{ik} \frac{\partial Q(t)}{\partial a_{ik}(t)} + \lambda^{(5)}_{ik} q_{ik}(t)$$  (9)

where $0 < \lambda^{(5)}_{ik} < 1$.

3 Properties of Generalized Particle Model

We summarize properties of GPM for enterprise computing optimization in the following Lemmas and Theorems, which involve the correctness, convergence and stability of GPM. For page limitation, their proofs are omitted.

Lemma 1. The first and second terms of Eqs. (8); (9) enable the particle $s_{ik}$ to increase the personal utility of the resource supplier $A_i$ from the user $G_k$, in direct proportion to $\lambda^{(1)}_{ik} + a \lambda^{(2)}_{ik}$.

Lemma 2. Updating $p_{ik}$ and $a_{ik}$ by Eqs.(8); (9); respectively, gives rise to monotonic increase of the aggregate utility of all the particles, in direct proportion to $a \lambda^{(2)}_{ik}$.

Lemma 3. If $\varepsilon$ is very small, then decreasing the potential energy $P(t)$ of Eq. (3) amounts to increasing the minimal utility of all the particles.

Lemma 4. The third terms of Eqs. (8); (9) enable the particle $s_{ik}$ to increase the minimal utility of all the particles, in direct proportion to $\lambda^{(3)}_{ik} w_{ik}^2(t)$, where

$$w_{ik}^2(t) = \exp[-(u_{ik}(t))^2 / 2\varepsilon^2] / \sum_{i=1}^{m} \exp[-u_{ik}(t)^2 / 2\varepsilon^2]$$

Lemma 5. The fourth terms of Eqs.(9); (10) enable the particle $s_{ik}$ to monotonic decrease of the potential energy $Q(t)$, in direct proportion to the value of $\lambda^{(4)}_{ik}$.
**Theorem 1.** Updating $p_{ik}$ and $a_{ik}$ by Eqs.(8); (9); respectively, gives rise to decreasing the hybrid energy function $\Gamma_{ik}(t)$ where very particle may autonomously determine its optimization objective according to its own personality and intention.

**Theorem 2.** The algorithm GPMA can dynamically optimize in parallel the resource allocation for enterprise computing in the context of multi-type social coordination, multi-degree autonomy and multi-objective optimization.

**Lemma 6.** If $\gamma - 1 > -\Psi^{(2)}_{ik}(t) > 0$, $\frac{\partial \Psi^{(2)}_{ik}(t)}{\partial q_{ik}(t)} < 1$ for $q_{ik}(t) < 0$ and $q_{ik}(t) > 1$; and $\frac{\partial \Psi^{(2)}_{ik}(t)}{\partial q_{ik}(t)} \geq 1 - \gamma$ for $0 < q_{ik}(t) < 1$ remain valid, then a stable equilibrium point of the particle $s_{ik}$ will be either ($q_{ik}(t) > 1, v_{ik}(t) = 1$) or ($q_{ik}(t) < 0, v_{ik}(t) = 0$).

**Fig. 3.** When $\gamma > 1$, the possible equilibrium points of the dynamic status $v_{ik}(t)$ of a particle $s_{ik}$. The point where $-\Psi^{(2)}_{ik}(t)$ equals $\Psi^{(1)}_{ik}(t)$ is an equilibrium point. The symbols $\ast, \circ, \|$ denote a stable equilibrium point, saddle point and unstable equilibrium point, respectively.

**Lemma 7.** If $\gamma > 1$, $-\Psi^{(2)}_{ik}(t) < 0$ and $\frac{\partial \Psi^{(2)}_{ik}(t)}{\partial q_{ik}(t)} < 1$ for $q_{ik}(t) > 1$ remain valid, then a stable equilibrium point of the particle $s_{ik}$ will be ($q_{ik}(t) > 1, v_{ik}(t) = 1$).
Lemma 8. \( \frac{\partial \psi''_k(t)}{\partial q_k(t)} < 1 \) for \( q_k(t) < 0 \) remain valid, then a stable equilibrium point of the particle \( s_k \) will be \( (q_{ik}(t) < 0, v_{ik}(t) = 0) \).

Lemma 9. If \( \gamma > 1, \frac{\partial \psi''_k(t)}{\partial q_k(t)} < 1 \) for \( q_k(t) = 1^0 \) and \( \frac{\partial \psi''_k(t)}{\partial q_k(t)} \geq 1 - \gamma \) for \( q_k(t) = 1^0 \) remain valid, then the equilibrium point \( (q_{ik}(t) = 1, v_{ik}(t) = 1) \) is saddle point. Moreover, if \( \gamma > 1, \frac{\partial \psi''_k(t)}{\partial q_k(t)} < 1 \) for \( q_k(t) = 1^0 \) and \( \frac{\partial \psi''_k(t)}{\partial q_k(t)} \geq 1 - \gamma \) for \( q_k(t) = 1^0 \) remain valid, then the equilibrium point \( (q_{ik}(t) = 0, v_{ik}(t) = 0) \) is saddle point.

Theorem 3. If \( \gamma > 1 \) and \( 0 \leq q_{ik}(t_0) \leq 1 \) remain valid, then the dynamical Eq. (6) has a stable equilibrium point if \( 0 < -\Psi^{(2)}_{ik}(t) < \gamma - 1 \).

Theorem 4. If the condition of Theorem 3 remains valid, then GPM will converge to a stable equilibrium state.

4 Simulations

Some simulation results on the algorithm GPMA for the enterprise computing optimization in the context of resource allocation and task assignment in complex environment are given as follows.
Fig. 4. For different problem sizes, the transient of minimal personal utility among all the particles during executing the algorithm GPMA, where the number of particles: 100, 400, 900, 1600 corresponds to problem size: \(10 \times 10\); \(20 \times 20\); \(30 \times 30\); \(40 \times 40\), respectively.

- **Influence of problem size on the utility for enterprise computing optimization:** For different problem sizes, the transients of the minimal personal utility among all the particles and the aggregate utility of all the particles during executing the algorithm GPMA are shown in Fig. 4 and Fig. 5, respectively. We can see that for different problem sizes using the GPMA always gives rise to the increase of the minimal personal utility.

![Graph showing utility vs. iteration for different problem sizes](image)

Fig. 5. For different problem sizes, the transient of the aggregate utility of all the particles during executing the algorithm GPM.

- **The influence of problem size on optimization criteria:** In order to evaluate the optimality performance of GPMA, we use the three criteria: the fairness FN, resource utilization rate RUR, and user satisfactory degree USD. For different problem sizes, the transients of the aggregate utilization rate of resource suppliers, and aggregate satisfactory degree of resource users are shown in Fig. 6.

![Graph showing RUR and USD vs. iteration for different problem sizes](image)
Fig. 6. For different problem sizes, the transients of the aggregate utilization rate of resources and the aggregate satisfactory degree of users during executing the algorithm GPMA.

![Graph showing transient behavior](image)

Fig. 7. For $30 \times 30$ problem size, the performance comparison between GPAA and the Max-Min Algorithm in terms of transients of the allocation fairness, aggregate utilization rate of resource suppliers, and aggregate satisfactory degree of resource users.

- **Comparisons:** The comparisons between the algorithm GPMA and the famous Max-Min algorithm (MMA) are shown in Fig. 7, which demonstrate that, for different problem sizes, GPMA can converge to a stable equilibrium solution much faster than the MMA. The algorithm GPMA exhibits much better optimality performance than MMA in terms of the aggregate utilization rate of resource suppliers and aggregate satisfactory degree of resource users, whereas they have almost approximately equal allocation fairness.

5 Conclusions

We have proposed a new generalized particle model (GPM) for parallel and distributed optimization of enterprise computing in complex environment. The GPM's properties have been proven in detail. GPM may deal with multi-type social coordination, multi-degree autonomy, multi-objective optimization, multi-granularity coalition, and some complex phenomena, e.g. congestion, failure and priority. The proposed generalized particle approach also has the advantages in terms of parallelism and feasibility for hardware implementation by VLSI technology.
References