We Claim:

1. A method to schedule virtual machine (VM) migrations, the method comprising:
   analyzing VM migration behavior;
   building a simulation tool to predict time for multiple migrations under different links conditions and VM characteristics;
   determining a predetermined bandwidth sharing policy for each network link;
   applying a bin-packing technique to organize bandwidth resources from all network links; and
   allocating the links to different migration tasks.

2. The method as claimed in claim 1, comprising simulating the performance of a single VM migration.

3. The method as claimed in claim 1, comprising simulating the performance of a plurality of VM migrations.

4. The method as claimed in claim 3, comprising:
   performing network resource multiplexing; and
   determining overhead associated with communication protocols.

5. The method as claimed in claim 1, comprising determining global resource assignments.

6. The method as claimed in claim 5, comprising determining a network link capacity.

7. The method as claimed in claim 5, comprising packing multiple migrations into one.

8. The method as claimed in claim 7, comprising:
   performing migration demand estimation; and
   packing each item into a bin representing all network links wherein the item denotes each migration task.

9. The method as claimed in claim 1, comprising determining available bandwidth in the network links as

$$\mathsf{C}^{(new)} = \mathsf{C} - \sum_{active\ migrations\ M_k} D^{(k)}$$

where C is the original link capacity described in equation (6) and $D^{(k)}$s are the bandwidth demands of ongoing migrations.

10. The method as claimed in claim 1, comprising minimizing VM migration time by minimizing a *Premigration Overhead*, time spent in the `iterative pre-copy' and `stop-and-copy' stages, and a *Postmigration Overhead*.

11. The method as claimed in claim 1, comprising determining end-to-end bandwidth demand for migration M_i as a vector

$$\mathsf{D}^{(i)} = \mathsf{P}^{(i)} \times d^{(i)} = [1,0,0,\cdots,1]^{\mathrm{T}} \times d^{(i)}$$

Where a binary vector $\mathsf{P}^{(i)} = [1,0,0,\cdots,1]^{\mathrm{T}}$ represents a path of migration M_i with a value `1' in the $i$th entry indicating an inclusion of the $i$th link in the path and `0' vice versa and where $d^{(i)}$ is an expected bandwidth allocated to M_i.

12. The method as claimed in claim 1, comprising determining a local bandwidth demand $d_j^{(i)}$ of M_i in link $l_j^{(i)}$ as $d_j^{(i)} = c_j / s_j^{(i)}$.

13. The method as claimed in claim 1, comprising:

transforming a resource demand vector $\mathsf{D}^{(i)}$ for each migration M_i into a scalar $\eta^{(i)}$, $\mathsf{D}^{(i)} \rightarrow \eta^{(i)}$, where $\eta^{(i)}$ equals to a summation of all the elements in $\mathsf{D}^{(i)}$.

sorting resource demands based on their transformed $\eta^{(i)}$ values ; and

scanning migration tasks from high to low $\eta^{(i)}$s, wherein for each selected migration task, placing the selected task in an earliest time interval T with free capacities to host the task.

14. The method as claimed in claim 1, comprising

determining bandwidth demand $d_j^{(i)}$ for all links $l_j^{(i)}$, $j = 1, \cdots, k$ along M_i's path; and

determining an overall bandwidth demand of M_i as a maximum demand among all the local estimations $d^{(i)} = \max\left\{d_1^{(i)}, d_2^{(i)}, \cdots, d_k^{(i)}\right\}$.

15. A system to schedule virtual machine (VM) migrations, the system comprising:

means for analyzing VM migration behavior;

means for building a simulation tool to predict time for multiple migrations under different links conditions and VM characteristics;

means for determining a predetermined bandwidth sharing policy for each network link;

means for applying a bin-packing technique to organize bandwidth resources from all network links; and

means for allocating the links to different migration tasks.

Dated this 25th day of April, 2014

[R. MAHESH]
OF REMFRY & SAGAR
ATTORNEY FOR THE APPLICANT(S)