



- (51) **International Patent Classification:**
G06F 9/48 (2006.0 1) G06F 9/46 (2006.0 1)
- (21) **International Application Number:**
PCT/US20 15/057494
- (22) **International Filing Date:**
27 October 2015 (27.10.201 5)
- (25) **Filing Language:** English
- (26) **Publication Language:** English
- (71) **Applicant: HEWLETT PACKARD ENTERPRISE DEVELOPMENT LP** [US/US]; Hewlett Packard Enterprise Development LP, 11445 Compaq Center Drive West, Houston, Texas 77070 (US).
- (72) **Inventors: CHUA, Freddy;** 1501 Page Mill Road, Palo Alto, California 94304 (US). **HUBERMAN, Bernardo;** 1501 Page Mill Road, Palo Alto, California 94304 (US).
- (74) **Agents: PAGAR, Preetam B.** et al; Hewlett Packard Enterprise, 3404 East Harmony Road, Mail Stop 79, Fort Collins, Colorado 80528 (US).
- (81) **Designated States** (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY,

BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

- (84) **Designated States** (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

- as to the identity of the inventor (Rule 4.1 7(I))
- as to applicant's entitlement to apply for and be granted a patent (Rule 4.1 7(H))

[Continued on nextpage]

- (54) **Title:** PARTITIONING A WORKFLOW

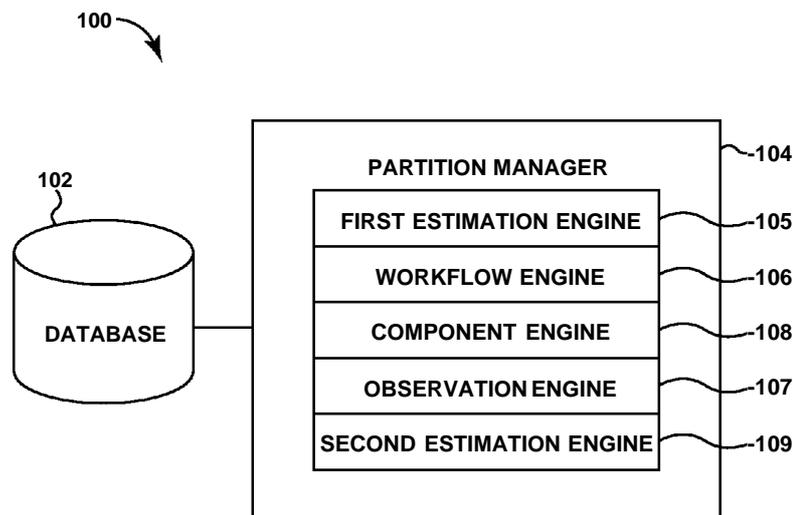


Fig. 1

(57) **Abstract:** Example implementations relate to partitioning a workflow. An expected completion time and a corresponding expected variance of the expected completion time of an execution of a portion of a workflow for each of a plurality of executors are determined based on a corresponding execution speed of the executor and a property of the portion of the workflow. A partition of the workflow into workloads is to be allocated among the plurality of executors that achieves a workflow completion time under a threshold completion time, and the workloads to each of the plurality of executors are allocated according to the determined partition.



Published:

— with international search report (Art. 21(3))

PARTITIONING A WORKFLOW

Background

[0001] Workflows may be partitioned into smaller workloads. Partitioning of workloads may accelerate the completion of the full workflow. For example, parallel execution of workloads may cause a more rapid completion of a workflow than a serial execution of the workloads. A workflow, or task, is considered complete when both independent components are complete, with the completion time taken as the maximum of the two components. Each component operates on distinct processing units with different configurations and capabilities, so each component may have a completion time that follows a different statistical distribution.

Brief Description of the Drawings

[0002] Figure 1 illustrates an example system, according to the disclosure;

[0003] Figures 2A-2C illustrate example graphs, according to the disclosure;

[0004] Figures 3A-3B illustrate example graphs, according to the disclosure;

[0005] Figures 4A-4B illustrate example graphs, according to the disclosure;

[0006] Figure 5 illustrates an example of a computing device, according to the disclosure;

[0007] Figure 6 illustrates a flow chart of an example of a method, according to the disclosure; and

[0008] Figure 7 illustrates a flow chart of an example of a method, according to the disclosure.

Detailed Description

[0009] A workflow can be any job, task, and/or set of operations. An example of a workflow can be a large file to be transmitted over the Internet, a large print job capable of being executed on more than one printer, a manufacturing process capable of being broken into different streams, traffic capable of being routed over more than one route, and/or a number of other job, tasks, and/or set of operations capable of execution on a plurality of executors.

[0010] An executor can be any mechanism capable of executing the workflow or a workload portion thereof. For example, an executor can be a processor of a computing device, a virtual machine, a printer, a manufacturing stream, a traffic route, and/or a number of other mechanisms by which to execute a portion of a workflow.

[0011] An executor can have a property associated therewith. For example, the executor can have an associated execution speed. An execution speed can be a rate of performing the workflow and/or a workload portion thereof. For example, an execution speed can be a processing speed of a processor of a computing device, an operations rate of a virtual machine, a page printing rate of a printer, a widget producing rate of a manufacturing stream, a traffic capacity and/or traffic speed associated with a traffic route, and/or any other metric related to the speed at which an executor is capable of executing a workflow.

[0012] Executors can be heterogeneous. Executors can have different properties, such as execution speed, from one another. Additionally, each executor can possess a distinct amount of uncertainty or variance associated with its execution speed. This variance can be a result of fluctuations in the properties and/or characteristics of the executor and/or demands placed on the executor and/or its resources.

[0013] Partitioning a workflow into workloads allocated among heterogeneous executors can be accomplished through equal partitioning of the workflow. However, this can lead to a workflow reaching completion at an undesired time; for instance upon completion by the slowest executor of its workload portion of the workflow.

[0014] In contrast, examples herein relate to determining an expected completion time for a workflow and an expected corresponding variance of the completion time for the workflow associated with each of a number of partitions of the workflow among a number of executors each executing a respective workload portion of the workflow in parallel. The workflow can then be partitioned into workloads allocated among the number of executors based on the determined expected completion time for the workflow and the determined expected corresponding variance of the completion time for the workflow associated with each of the number of partitions of the workflow. That is, in contrast to assuming homogeneity among executors and equally dividing a workflow into equally sized workloads for executors, examples herein relate to partitioning the workflow into workloads based on expected completion times and corresponding variance thereof associated with the partition.

[0015] The parallelization procedure examples herein include a decision on how to partition a workflow so that its full completion process takes the shortest amount of time to complete with a relatively minimal amount of uncertainty and/or variance in the completion time. Uncertainty and/or variance are relevant variables because, for example, unavoidable fluctuations in executing a workload that each executor (e.g., processing unit, channel, virtual machine, etc.) undergoes when having to time share with other processes and/or workflows. This can introduce a stochastic component into the execution of any workflow, which can at times increase the amount of time it takes for a given workflow and/or workload portion of a workflow to complete. Therefore, incorporating these fluctuations into a partitioning procedure can result in shorter times to completion than an original non-partitioned workflow.

[0016] The parallelization procedure can include partitioning a workflow into a plurality of workloads in such a way that their outputs are

joined and their full completion time takes less time and uncertainty and/or variance than when running it in one executor. The parallelization procedure can be implemented to affect a partitioning for joint parallel execution of workload portions of a workflow with an implication that the overall execution time of the workflow is determined by the longest running executor.

[0017] For instance, examples in accordance with the present disclosure may be directed to estimating processing characteristics of a processing unit. A variant of the Gibbs sampling method (e.g., algorithm) can be used to learn processing characteristics of heterogeneous parallel processing unit(s) so that a decision can be made with respect to what value f should take, where f is a proportion for processing in parallel. For instance, large and time-consuming tasks can be broken down into two independent workloads, with proportions f and $1-f$ for processing in parallel.

[0018] Figure 1 illustrates an example of a workflow partition system 100, according to examples of the present disclosure. As illustrated in Figure 1, the workflow partition system can include a database 102, a partition manager 104, a number of engines 105, 106, 107, 108, and 109. The partition manager 104 can include the number of engines 105, 106, 107, 108, and 109 in communication with the database 102 via a communication link. The system can include additional or fewer engines than illustrated to perform the various tasks described herein.

[0019] The number of engines 105, 106, 107, 108, and 109 can include a combination of hardware and instructions (e.g. programming) which are executed by the hardware to perform tasks including those described herein (e.g., estimate processing capabilities of system components, partition a workflow associated with the plurality of system components into a plurality of workloads, etc.). The instructions can include instructions (e.g., software (e.g., instructions that can be executed by hardware such as a processor), hardware with instructions, etc.) stored in a memory resource (e.g., computer readable medium (CRM), machine readable medium (MRM), etc.) as well as hard-wired program (e.g., logic).

[0020] The first estimation engine 105 can include hardware and/or a combination of hardware and instructions (e.g., programming) which are executed by the hardware to estimate a processing capability of each of a plurality of system components. For instance, a workflow can be received by the first estimation engine, and an estimation can be made as to the processing capabilities of components associated with the system and the workflow. The processing capabilities can be workload completion times of each of the plurality of workloads. The first estimation engine may estimate the processing capability of each of the plurality of system components using a Bayesian approach, as will be discussed further herein. The workflow can include real workflows processing in parallel. As used herein, components are entities that process workloads. For instance, depending on a workflow, a component can be a central processing unit (CPU), among others.

[0021] The workflow engine 106 can include hardware and/or a combination of hardware and instructions (e.g., programming) which are executed by the hardware to perform tasks to partition the workflow associated with the plurality of system components into a plurality of workloads. The workflow can be partitioned randomly. As used herein, randomly partitioning includes an inconsistent splitting of the workflow. For instance, the partition may be 50/50, 40/60, 55/45, or any other combination between two workloads. A partition between two workloads is used as an example here, but the partition may be into more than two workloads.

[0022] The partition can reduce an expected workflow completion time and reduce an expected variance corresponding to the expected workflow completion time. That is, the workflow engine 106 can include hardware and/or a combination of hardware and instructions (e.g., programming) to perform tasks to measure statistical properties of a workflow allocation system and determine a partition of the workflow based on the measure of statistical properties such that an expected workflow completion time and an expected variance corresponding to the expected workflow completion time are reduced.

[0023] The component engine 108 can include hardware and/or a combination of hardware and instructions which are executed by the hardware to run each of the plurality of workloads through each of the plurality of components, and the observation engine 107 can include hardware and/or a combination of hardware and instructions which are executed by the hardware to evaluate how each of the plurality of components handles each of the plurality of workloads. For example, the observation engine can determine how long each workload takes to complete in each component. The second estimation engine 109 can include hardware and/or a combination of hardware and instructions which are executed by the hardware to update the processing capability estimates of each of the plurality of system components based on evaluation. For instance, as will be discussed in further detail below, the system can "learn" how to better partition the workflow.

[0024] Determining the partition can include determining an expected workflow completion time associated with each of a plurality of partitions of the workflow among a plurality of parallel executors of the workflow. That is, determining the expected workflow completion time can be based on a rate of performing a workload and a property of the workload. The expected completion time for the workflow can be determined based on the rate of performing the workload and a property of the workload. The rate of performing the workflow can include an execution speed associated with one or more of the executors. The execution speed can be a known stored value (e.g., in database 102) and/or can be measured across repeated trials of a partitioning process conducted over a period of time utilizing various partitions. The property of the workload can include a metric of how intensive the workload will be on an executor/executors. A metric of how intensive the workload will be can include a size of the workflow/workload, a time associated with executing the workflow/workload, a complexity of the workflow/workload, an amount of resource demand associated with executing the workflow/workload, an amount of operations involved in executing the workflow/workload. The expected completion time can additionally be

determined based on the expected corresponding variance of the completion time for the workflow.

[0025] Determining the partition can further include determining an expected variance corresponding to each expected workflow completion time. That is, determining the expected workflow completion time and an expected corresponding variance of the completion time can include numerically deriving values of the expected completion time for a workflow and an expected corresponding variance of the completion time for each of a number of potential partitions of the workflow. The expected workflow completion time and an expected corresponding variance of the workflow completion time can be determined based on the fluctuations in an execution speed associated with each of the plurality of parallel executors. Parallel executors can include executors that separately perform the joint execution of workload portions of a workflow in a parallel. The parallel executors can be heterogeneous in that they each are associated with distinct execution speeds and fluctuations of those execution speeds brought on by characteristics of the executor, characteristics properties of the executor resources, and/or competition/depletion of executor resources, among other demands. The expected variance of the workflow completion time can be a numerical categorization of the fluctuation in the workflow and/or workload completion time brought on by the distinct fluctuations associated with utilizing particular executors to execute particular workload portions of the workflow.

[0026] As described above, the expected variance corresponding to each expected workflow completion time can be determined based on fluctuations in the execution speed associated with the each number of executors executing a respective workload portion of the workflow in parallel. These fluctuations can be stored as fluctuation values (e.g., in database 102) associated with each executor and/or be measured through repeated trials of a partitioning process conducted over a period of time utilizing various partitions and/or measured by inducing competition for an executor resource during test trials.

[0027] Further, the numerical values for the expected workflow completion time and an expected corresponding variance of the workflow

completion time can be derived according to a model of the executors of the workflow and how they can execute a workflow based on their properties. For example, as noted above, a workflow can be partitioned into two workloads i and j (while this example only characterizes the partitioning of the workflow into two workloads to be allocated across two executors the examples included herein are not so limited are equally applicable to additional partitions and additional executors), each of which can be executed on different executors, with different execution speeds and fluctuating performance parameters. The workloads can have proportions of f and $1-f$ for processing in parallel. The workflow can be considered complete when both independent components complete, with the completion time taken as the maximum of the two workloads. Given that each workload operates on distinct executors (e.g., processing units) with different configurations and capabilities, each component has a completion time that follows a different statistical distribution.

[0028] With Θ_i representing the parameters of i 's completion time t_i , and Θ_j representing the parameters of j 's completion time t_j , the probability that the task has a completion time t before e is given by:

$$P(t \leq e|f, \Theta) = P(t_i \leq e|f, \Theta_i) \cdot P(t_j \leq e|f, \Theta_j)$$

$$Q = \{Q_i, Q_j\}.$$

[0029] The expected completion time of the task t is given by:

$$E(t|f, \Theta) = \int_0^\infty 1 - P(t \leq \epsilon|f, \Theta) d\epsilon,$$

while the variance of completion time t is given by:

$$Var\{t|f, \Theta\} = \int_0^\infty \epsilon^2 [1 - P(t \leq \epsilon|f, \Theta)] d\epsilon - [E(t|f, \Theta)]^2.$$

[0030] For brevity, the expected completion time $E(t|f, \Theta)$ can be denoted as $\mu(i)$ and the variance of completion time $Var\{t|f, \Theta\}$ as $\sigma^2(f)$. For the purpose of illustration as to how $\mu(i)$ and $\sigma^2(f)$ varies as a function of f , it can be assumed that the completion times on each processing unit is Gaussian, with known values of Θ , which governs the processing capabilities of the two processing units. By using the hypothetical values $\mu_i = 30$, $\sigma_i = 2$, $\mu_j = 20$, $\sigma_j = 6$, the numerical results as shown in Figures 2A, 2B, and 2C can be obtained.

[0031] Figures 2A-2C illustrate example graphs, according to the disclosure.

Figure 2A illustrates $\mu(i)$ (y-axis 271-1) and $\sigma^2(f)$ (x-axis 269-1) for each value of f . Figure 2B illustrates $\mu(i)$ (y-axis 271-2) with respect to f (x-axis 273-1), and Figure 2C illustrates $\sigma^2(f)$ (y-axis 269-2) with respect to f (x-axis 273-2). In Figure 2A, each point gives the respective value of $\mu(i)$ and $\sigma^2(f)$ for a specific f . The curve formed from these points is parabolic which indicates that for some values of $\mu(i)$, there are two possible choices of $\sigma^2(f)$. The converse is true as well, for example, some values of $\sigma^2(f)$ have two choices of $\mu(i)$. If the assumptions on the statistical distribution of completion times for the two parallel machines holds, then the theoretical results derived in Figure 2A allow a decision on an appropriate value of f that minimizes the expected time $\mu(i)$ for a desired amount of uncertainty $\sigma^2(f)$ and minimizes the amount of uncertainty $\sigma^2(f)$ for a desired expected time $\mu(i)$.

[0032] The appropriate choice of f that provides the optimal values of $\mu(i)$ and $\sigma^2(f)$ is given by the efficient frontier 270-1 (as will be discussed further herein) in the lower left portion of the curve. f thus denotes the amount of appropriate parallelism necessary to achieve a desired level of Quality-of-Service (QoS). The scenarios that find QoS important may include supply chain management, computer networking, parallel and distributed systems, or military strategies that often require the achievement of a common objective orchestrated by several teams working in parallel, among others.

[0033] In some examples, knowledge of a processing unit may be vague or unknown. Other approaches may include addressing this issue by performing many controlled experiments of different workloads by varying f , with each f having multiple numbers of trials in order to estimate the mean and variance at each value of f . However, in realistic or deployed systems, such controlled experiments represent an opportunity cost and the resources used to conduct such experiments would reap more benefits by running actual workloads.

[0034] As such, examples of the present disclosure use a method (e.g., algorithm) that learns the (processing) system parameters based on

trials of using the processing units without deliberate selection off. A Bayesian approach may be used to infer the parameters Θ . Using a Bayesian approach may allow for a current understanding of a system's performance given as input to the method using prior beliefs expressed using statistical distributions. Also, based on an observed batch of data, such as completion time with respect to the amount of parallelism f , the likelihood of the observations can be combined with the prior beliefs to obtain a posterior belief of the systems' performance. In addition, a posterior belief obtained from the previous batch of observations can become the prior belief for the next batch of observations. By chaining a sequence of prior and posterior updates, the method can adjust the system's parameters for a dynamically changing environment.

[0035] For simplification (but without loss of generality) of the learning algorithm description, we shall assume that the completion time t_i, t_j for each of the processing unit i, j follows that of a Normal distribution:

$$p(t_i | f, \mu_i, \sigma_i, \alpha_i, \beta_i) \sim N(f^{\alpha_i} \mu_i, [f^{\beta_i} \sigma_i]^2)$$

$$p(t_j | 1 - f, \mu_j, \sigma_j, \alpha_j, \beta_j) \sim N((1 - f)^{\alpha_j} \mu_j, [(1 - f)^{\beta_j} \sigma_j]^2)$$

[0036] The notation may be simplified, so that:

$$f_i = f$$

$$f_j = 1 - f_i.$$

[0037] As such, the analysis for i and j is identical:

$$p(t_i | f_i, \mu_i, \sigma_i, \alpha_i, \beta_i) \sim N(f_i^{\alpha_i} \mu_i, f_i^{2\beta_i} \sigma_i^2)$$

$$p(t_j | 1 - f_j, \mu_j, \sigma_j, \alpha_j, \beta_j) \sim N((1 - f_j)^{\alpha_j} \mu_j, (1 - f_j)^{2\beta_j} \sigma_j^2).$$

[0038] The purpose of simplifying for i and j is to show that if the Bayesian updates of i can be derived, then similarly, the same equations can be applied to j . With that, clutter can be reduced in the equations by dropping the subscripts i and j leaving the following:

$$p(t | f, \mu, \sigma, \alpha, \beta) \sim N(f^\alpha \mu, f^{2\beta} \sigma^2). \quad (1)$$

[0039] As stated in equation 1, the completion time t can be predicted conditioned on the assumption that $\mu, \sigma, \alpha, \beta$ are known.

[0040] As noted, Bayesian inference equations can be used to obtain estimates for the values of μ , σ , a , and β . Since μ and σ are unknowns, it can be assumed that they are drawn from some statistical distributions. For notational convenience, the variance σ^2 can be replaced with the precision λ using the following relationship:

$$\lambda = \frac{1}{\sigma^2}.$$

[0041] An appropriate choice of prior distribution for μ is the following Normal distribution :

$$\begin{aligned} \mu | \mu_0, k_0, \lambda &\sim N(\mu_0, (V \lambda)^{-1}) \\ p(\mu | \mu_0, k_0, \lambda) &\propto \lambda^{1/2} \exp\left(-\frac{k_0 \lambda}{2} (\mu - \mu_0)^2\right). \end{aligned}$$

[0042] While the prior distribution of λ is the Gamma distribution:

$$\begin{aligned} \lambda | v_0, \psi_0 &\sim \text{Gamma}(v_0, \text{rate} = \psi_0) \\ p(\lambda | v_0, \psi_0) &\propto \lambda^{v_0-1} \exp(-\psi_0 \lambda), \end{aligned}$$

where μ_0 , k_0 , v_0 , and ψ_0 are parameters for the prior distributions of μ and λ , which are constants that can be set based on subjective prior knowledge.

[0043] Then, expressing the probability density function of the posterior distribution function as a multiplication of the two distributions can be given as:

$$\rho(\mu, \lambda | \mu_0, \kappa_0, v_0, \psi_0) \propto \lambda^{v_0 - \frac{1}{2}} \cdot \exp\left(-\frac{\lambda}{2} [\kappa_0 (\mu - \mu_0)^2 + 2\psi_0]\right) \quad (2)$$

$$\rho(\mu, \lambda | T, F, \alpha, \beta, \mu_0, \kappa_0, v_0, \psi_0) \propto \rho(T | F, \mu, \lambda, \alpha, \beta) \cdot \rho(\mu, \lambda | \mu_0, \kappa_0, v_0, \psi_0) \quad (3)$$

[0044] The likelihood is then given by:

$$p(T | F, \mu, \lambda, \alpha, \beta) \propto \frac{\lambda^{\frac{N}{2}}}{\prod_n f_n^\beta} \exp\left(-\frac{\lambda}{2} \sum_n \left[\frac{t - f_n^\alpha \mu}{f_n^\beta}\right]^2\right) \quad (4)$$

$$\propto \lambda^{\frac{N}{2}} \exp\left(-\frac{\lambda}{2} \sum_n \left[\frac{t - f_n^\alpha \mu}{f_n^\beta}\right]^2\right) \quad (5)$$

[0045] Equations 2 and 5 can be substituted into equation 3, and through algebraic manipulations (e.g., expansion, completing the square, factorization and simplification), the following can be obtained, and the posterior distribution given by:

$$p(\mu, \lambda | T, F, \alpha, \beta, \mu_0, \kappa_0, \nu_0, \psi_0) \propto \lambda^{\nu_N - \frac{1}{2}} \cdot \exp\left(-\frac{\lambda}{2} [\kappa_N (\mu - \mu_N)^2 + 2\psi_N]\right).$$

$$\mu_N = \frac{\mu_0 \kappa_0 + \sum_n f_n^{\alpha-2\beta} t_n}{\kappa_0 + \sum_n f_n^{2\alpha-2\beta}}$$

$$\kappa_N = \kappa_0 + \sum_n f_n^{2\alpha-2\beta}$$

$$\nu_N = \nu_0 + \frac{N}{2}$$

$$\psi_N = \psi_0 + \frac{1}{2} \left[-\mu_N^2 \kappa_N + \mu_0^2 \kappa_0 + \sum_n \left(\frac{t_n}{f_n^\beta} \right)^2 \right]$$

[0046] α and β represent the scalability of the processing unit when given different workloads governed by f . A perfect system would have $\alpha = 1.0$ and $\beta = 1.0$ indicating that the expected completion time and variance scales linearly with the size of the workload. However, $\alpha > 1.0$ and $\beta > 1.0$ may represent an impossible scenario since this suggests that the system takes less time and has less uncertainty when given more workload.

[0047] Since α and β may only take values between 0 and 1, it may be appropriate to use the Beta distribution as the prior of α and β :

$$p(\alpha | \theta_0, \phi_0) \propto \alpha^{\theta_0-1} (1 - \alpha)^{\phi_0-1}$$

$$p(\beta | \delta_0, \eta_0) \propto \beta^{\delta_0-1} (1 - \beta)^{\eta_0-1}$$

[0048] Using the likelihood given by Equation 5, the posterior distribution of α conditioned on a set of observations T for a given set of F is:

$$p(\alpha | T, F, \mu, \lambda, \theta_0, \phi_0, \beta) \propto p(T | F, \mu, \lambda, \alpha, \beta) p(\alpha | \theta_0, \phi_0) \propto \lambda^{\frac{N}{2}} \exp\left(-\frac{\lambda}{2} \sum_n \left[\frac{t_n - f_n^\alpha \mu}{f_n^\beta} \right]^2\right) \alpha^{\theta_0-1} (1 - \alpha)^{\phi_0-1} \quad (6)$$

[0049] For the posterior distribution of β , the likelihood given by Equation 4 may be used, which gives the following:

$$p(\beta | T, F, \mu, \lambda, \delta_0, \eta_0, \alpha) \propto p(T | F, \mu, \lambda, \alpha, \beta) p(\beta | \delta_0, \eta_0) \propto \frac{\lambda^{\frac{N}{2}}}{\prod_n f_n^\beta} \exp\left(-\frac{\lambda}{2} \sum_n \left[\frac{t_n - f_n^\alpha \mu}{f_n^\beta} \right]^2\right) \beta^{\delta_0-1} (1 - \beta)^{\eta_0-1} \quad (7)$$

$$\theta_N = E(\alpha) \left[\frac{E(\alpha)[1-E(\alpha)]}{Var(\alpha)} - 1 \right] \quad (8)$$

$$\phi_N = [1 - E(\alpha)] \left[\frac{E(\alpha)[1-E(\alpha)]}{Var(\alpha)} - 1 \right] \quad (9)$$

$$\delta_N = E(\beta) \left[\frac{E(\beta)[1-E(\beta)]}{Var(\beta)} - 1 \right] \quad (10)$$

$$\eta_N = [1 - E(\beta)] \left[\frac{E(\beta)[1-E(\beta)]}{Var(\beta)} - 1 \right] \quad (11)$$

[0050] Then, using the standard definitions for $E(a)$ and $Var(a)$ to derive their specific values:

$$E(a) = \int_0^1 \alpha \cdot \rho(\alpha | T, E, \mu, \lambda, \theta_0, \phi_0, \beta) da \quad (12)$$

$$E(a^2) = \int_0^1 a^2 \cdot \rho(\alpha | T, E, \mu, \lambda, \theta_0, \phi_0, \beta) da \quad (13)$$

$$Var(a) = E(a^2) - [E(a)]^2 \quad (14)$$

$$E(\beta) = \int_0^1 \beta \cdot p(\beta | T, F, \mu, \lambda, \delta_0, \eta_0, \alpha) d\beta \quad (15)$$

$$E(\beta^2) = \int_0^1 \beta^2 \cdot p(\beta | T, F, \mu, \lambda, \delta_0, \eta_0, \alpha) d\beta \quad (16)$$

$$Var(\beta) = E(\beta^2) - [E(\beta)]^2 \quad (17)$$

[0051] Figures 3A and 3B illustrate example graphs, according to the disclosure. For instance, Figures 3A and 3B illustrate an example of the differences between the true and approximate posterior distribution of a . Figures 4A and 4B illustrate example graphs, according to the disclosure. For instance, Figures 4A and 4B illustrate an example of the differences between the true and approximate posterior distribution of β .

[0052] Figures 3A and 4A include exact curves 375 and 474, respectively, while Figures 3B and 4B includes approximate curves 377 and 476, respectively. Equations producing curves 375 and 474 may not be used for Bayesian inferences. In order for the equations to be suitable for Bayesian inference, the posterior and prior must exhibit similar forms. As such, in accordance with the present disclosure, a method may be used to approximate an exact posterior equation in the form of the prior equation. The approximated posterior equation produces curves 377 and 476.

[0053] Equation 1 summarizes the use of the Bayesian inference equations for estimating the parameters of the processing system. In response to updating the parameters of the prior distributions, the distributions can be sampled instead of taking their mode or mean, so as

to avoid getting trapped in a local maxima of the log likelihood. Because the mean is close to the mode (as illustrated in Figures 3B and 4B), the sampling from their distributions may have a desired side effect of increasing the log likelihood of the overall system. An example Gibbs sampling of $\mu, \lambda, \alpha, \beta$ may be performed showing a convergence of the Gibbs sampling method presented in equation 1. It may illustrate that the Bayesian inference equations and the Gibbs sampling method may be used to estimate system parameters.

[0054] Returning to Figure 1, the workflow engine 106 and/or a receipt engine (not pictured) can include hardware and/or a combination of hardware and instructions which are executed by the hardware to partition the workflow into workloads according to the determined priority. For instance, in response to the estimations, a different workflow may be received, and the workflow partition may be updated. The workloads can be workloads allocated among the plurality of executors. The partition can be based on the determined partition which can be based on the determined (e.g., estimated) expected completion time associated with each of the plurality of partitions of the workflow and the determined expected variance corresponding to each expected work flow completion time. For example, as illustrated in Figures 2A-2C, utilizing the above stated models the expected completion time $\mu(f)$ and expected corresponding variance of the completion time for the workflow $\sigma^2(f)$ can be numerically derived for different values of f representing various potential partitions of the workflow which can be graphed as μ with respect to each value of f and σ^2 with respect to each value of f . An efficient frontier 270-1, 270-2, 270-3 can be identified representing range of partition values of f which provide a targeted combination of μ and σ^2 values. For example, the efficient frontier can be a target value can be a minima of μ and σ^2 on these graphs, but since the value of f that provides a minimum point of μ and σ^2 can be different, then the efficient frontier can include the range of values of f which include at least both of these minima. This can provide a range of choices of workload size for partitioning the workflow to achieve the target combination of μ and σ^2 values for the workflow.

[0055] The efficient frontier can additionally be identified in a graph where μ and σ^2 are graphed parametrically as a function of each other resulting in a parabolic curve. The statistical distribution assumptions of the completion times for the two parallel workloads characterized in the above stated model can permit selection of an appropriate value of f which minimizes μ and σ^2 for the full workflow execution.

[0056] Therefore, determining the partition can include determining a partition of the workflow into workloads by selecting a particular partition value (e.g., f which minimizes both μ and σ^2 relative to other values of f) from a range of partition values, of the number of partitions, corresponding to a minima for the determined completion time for the workflow and a minima the determined variance of the completion time (e.g., the efficient frontier and/or the range of values including the minima of μ and σ^2).

[0057] Further examples of the application of the above stated workflow partition system 100 can include a parallel partitioning of a least squares error function which is quadratic and therefore convex. Although a map-reduce application can be applicable to such a partitioning, it does not produce the targeted solution. The input D to the convex function, in this example, can be partitioned into two workloads D_1 and D_2 . A classical operation can then be applied to each of the workloads D_1 and D_2 to obtain a globally optimized solution θ_1 and θ_2 based on each of their inputs. The solution θ for the original workflow D can be obtained as a linear combination of the solution from each of the workloads.

$$\theta = f \theta_1 + (1 - f) \theta_2$$

[0058] The parallel global algorithm can be processed on two executors (e.g., virtual machines, each one with a CPU core). The mean completion times μ and variance σ^2 can be obtained at each partition value of f by repeating a plurality of trials of the process over a period of time using different partitions (values of f). The mean completion times μ and variance σ^2 can be graphed as a function of f and a performance curve similar to the performance curve derived in the above examples can be obtained. A curve can be fitted to the observed behavior of mean

completion times μ and variance σ^2 across various partition values H_0 to identify an efficient frontier and/or a partition value H_0 to achieve a target mean completion time μ and variance σ^2 .

[0059] Another example of the application of the above stated workflow partition system 100 can include performing a file transmission of a fixed size file in parallel from a source node to a destination node over two network paths. Since the TCP network protocol may not allow fine grain control of how file packets travel through the Internet, an intermediate overlay can be created to redirect a fraction of the file packets through a different path. In this example, the source node can be hosted in New York City, while the destination node can be hosted in Singapore. A traceroute can be utilized to demonstrate that network packets can be transmitted through the west coast of the United States before reaching the destination in Singapore, which implies that the network packet from New York City route through the Pacific Ocean to Singapore.

[0060] In order to determine if utilizing an alternate route of sending file packets from New York through the Atlantic Ocean using Europe as the overlay before finally reaching Singapore is a more efficient network channel, another host can be created in London to act as an overlay to receive file packets from New York City and forward them to Singapore. A file can be partitioned into workloads based on different values of f and sent across the two different networks (e.g., Atlantic Ocean, Pacific Ocean). Disk I/O delays can be excluded to ensure that network transmissions times' contribution to completion times are analyzed alone.

[0061] To measure the completion time of the two parallel file transfers, the node at the destination can measure the time of the last packet (from either channel) and then subtract the time of the request for the first packet (e.g., from both channels).

[0062] In order to measure the mean completion times μ and variance σ^2 thereof, the file transfer can be repeated a number of times over a time period. For each trial, a randomized value of f can be used. The results obtained for this file transmission example can be consistent with the performance curve derived in the above examples.

[0063] Figure 5 illustrates an example computing device 520 according to examples of the disclosure. The computing device 520 can utilize instructions (software and/or hardware with instructions) hardware, and/or logic to perform a number of tasks including those described herein. The computing device 520 can be a combination of hardware and program instructions to share information. The hardware, for example, can include a processing resource 522 and/or a memory resource 524 (e.g., CRM, MRM, database, etc.).

[0064] A processing resource 522, as used herein, can include a processor capable of executing instructions stored by a memory resource 524. Processing resource 522 can be implemented in a single device or distributed across multiple devices. The program instructions (e.g., computer readable instructions (CRI)) can include instructions stored on the memory resource 524 and executable by the processing resource 522 to implement a desired task (e.g., partition the workflow into workloads allocated to each of a first plurality of executors associated with the workflow according to the determined partition value, etc.).

[0065] The memory resource 524 can be in communication with the processing resource 522 via a communication link (e.g., a path) 526. The communication link 526 can be local or remote to a machine (e.g., a computing device) associated with the processing resource 522. Examples of a local communication link 526 can include an electronic bus internal to a machine (e.g., a computing device) where the memory resource 524 is one of volatile, non-volatile, fixed, and/or removable storage medium in communication with the processing resource 522 via the electronic bus.

[0066] A module and/or modules 528, 530, and 532 can include CRI that when executed by the processing resource 522 can perform a number of tasks including those described herein. A number of modules 528, 530, and 532 can be sub-modules of other modules. For example, the completion module 528 and the value module 530 can be sub-modules and/or contained within the same computing device. In another example, the number of modules 528, 530, and 532 can comprise individual modules at separate and distinct locations (e.g., CRM, etc.).

[0067] Each of the number of modules 528, 530, and 532 can include instructions that when executed by the processing resource 522 can operate as a corresponding engine described herein. For example, the completion module 528 and value module 530 can include instructions that when executed by the processing resource 522 can operate as the workflow engine 106. The partition module 532 can include instructions that when executed by the processing resource 522 can operate as the receipt engine (not illustrated).

[0068] The completion module 528 can include instructions that when executed by the processing resource 522 can determine an expected completion time for a workflow and a corresponding expected variance of the completion time for the workflow across a plurality of partition values. In some examples this can be among a plurality of heterogeneous executors. The expected completion time of a workflow and the corresponding expected variance of the completion time of the workflow can be derived based on a size of the workflow, a portion of the workflow to be allocated to a particular executor, and an execution speed of the particular executor. That is, the expected completion time and expected variance can be derived by stored values (e.g., in database 102) of a size of a workflow, a workload portion of the workflow to be allocated to a particular executor, and an execution speed associated with the particular executor and/or values of a size of a workflow, a workload portion of the workflow to be allocated to a particular executor, and/or an execution speed associated with the particular executor obtained from trials with the executors handling various workloads. The expected completion time of the workflow can be determined by a derived amount of time involved in totally executing a workflow obtained from times involved in executing partitioned workloads of a given size on an executor with a given execution speed, and a given variance of that execution speed. The derived expected completion time and expected variance thereof can be expressed as a function of the partition at various values. The executors can be heterogeneous in that they each are associated with a distinct execution speed and/or a distinct variance thereof.

[0069] The value module 530 can include instructions that when executed by the processing resource 522 can determine a partition value of the plurality of partition values that produces a shortest expected completion time and a smallest corresponding expected variance of the completion time. For example value module 530 can determine a number of workflow partitions of the plurality of workflow partitions that achieve a target expected completion time of the workflow and a corresponding target expected variance of the completion time of the workflow. That is, the value module 530 can include instructions that when executed by the processing resource 522 can determine the number of workflow partitions of the plurality of workflow partitions that achieve the target values by application of the models described above in more detail with regard to Figure 1. That is, the value module 530 can include instructions that when executed by the processing resource 522 can compare the derived expected completion time of a workflow and the corresponding expected variance of the completion time of the workflow across a plurality of workflow partitions with a target value for each. For example, this can include deriving the expected completion time of a workflow and the corresponding expected variance of the completion time of the workflow as a function of the partition value and identifying a target range of values of partition values f including the minima of the expected completion time of a workflow as a function of f and/or the minima of the corresponding expected variance of the completion time of the workflow as a function of f .

[0070] The partition module 532 can include instructions that when executed by the processing resource 522 can partition the workflow into workloads allocated to each of a first plurality of executors associated with the workflow according to the determined partition value. The workloads may be executed in parallel by each of the number of executors according to a selected partition of the determined number of partitions. Partitioning the workflow can include splitting the workflow into workloads to be executed in parallel by each of the number of executors. The partition can be a partition value f selected from the target range of values of partition values f including the minima of the expected completion time of a

workflow as a function of f and/or the minima of the corresponding expected variance of the completion time of the workflow as a function of f . The selected partition value f_{can} be the partition that produces a shortest expected completion time with a smallest corresponding expected variance relative to the other derived expected completion times and corresponding expected variances of the completion time of other workflow partitions of the plurality of workflow partitions.

[0071] The partition module 532 can further include instructions that when executed by the processing resource 522 can determine an execution speed and a corresponding variance of the execution speed of each of a second plurality of executors and determine a property of a workflow associated with the second plurality of executors. The first plurality of executors may be heterogeneous, while the second plurality of executors may be homogeneous.

[0072] The partition module 532 can further include instructions that when executed by the processing resource 522 can piece completely executed workloads together to output a completed workflow.

[0073] Figure 6 illustrates a flowchart of an example method 640 according to examples of the disclosure. The example method 640 can be implemented using the system 100 illustrate in Figure 1 and/or the computing device 520 illustrated in Figure 5.

[0074] At 642, the method 640 can include determining an execution speed of a plurality of homogeneous executors and/or a corresponding variance thereof. Determining the execution speed and/or corresponding variance thereof can be accomplished by obtaining corresponding measurements throughout trials of the executor and/or from stored values (e.g., in database 102) associated with the executor.

[0075] At 644, the method 640 can include determining a property of a workflow associated with the plurality of homogeneous executors. Determining a property of the workflow can include determining a size, technical complexity, and/or resource demand of a particular workflow. Determining a property of the workflow can also include determining how a workflow may be partitioned into workloads and/or compatible executors to execute the workload.

[0076] At 646, the method 640 can include determining an expected completion time of a workflow and a corresponding expected variance of the completion time for the workflow across a plurality of partition values. That is, the method 640 can include numerically deriving an expected completion time of a workflow and an expected variance of the completion time of the workflow as a function of various partition values f corresponding to potential partitions of the workflow into workloads to be allocated to the plurality of heterogeneous executors.

[0077] At 648, the method 640 can include determining a partition value of the plurality of partition values that will produce a shortest expected completion time and a smallest corresponding expected variance of completion time as compared to the others associated with the plurality of partition values. The determined partition value can be a range of values including the minima of the expected completion time and the expected variance of completion time as a function of various partition values f . Alternatively, the partition value can be a single value of f corresponding to the shortest expected completion time along with a smallest expected variance of completion time.

[0078] At 650, the method 640 can include partitioning the workflow into workloads allocated to a plurality of heterogeneous executors according to the determined partition value (e.g., from 648 of the method 640).

[0079] At 652, the method 640 can include joining together the completed workloads upon completion by the plurality of heterogeneous executors to form a completed workflow.

[0080] Figure 7 illustrates a flowchart of an example method 780 according to examples of the disclosure. The example method 780 can be implemented using the system 100 illustrated in Figure 1 and/or the computing device 520 illustrated in Figure 5.

[0081] At 782, the method 780 can include determining an expected completion time and a corresponding expected variance of the completion time of an execution of a portion of a workflow for each of a plurality of executors based on a corresponding execution speed of the executor and a property of the portion of the workflow. The plurality of executors can be

heterogeneous executors. The execution speed of the executor can be determined based on estimates of the execution speed collected during deployment of the executor in executing workloads.

[0082] At 784, the method 780 can include determining a partition of the workflow into workloads to be allocated among the plurality of executors that achieves a workflow completion time under a threshold completion time. The determined partition of the workflow can be a partition of the workflow into workloads allocated to executors that produces a workflow completion time under a threshold completion time wherein the threshold completion time is a predetermined threshold completion times based on a Quality-of-Service level agreement. For example, the determined expected completion time and a corresponding determined expected variance of the completion time of an execution of a portion of a workflow for each of a plurality of heterogeneous executors can be utilized as the basis for formulating a pricing model for different levels of Quality-of-Service. The determined expected completion times and corresponding determined expected variances of the completion time of an execution of a portion of a workflow for each of a plurality of heterogeneous executors can be segmented into different pricing structures which a customer may select from. A selection of a particular pricing structure can correspond to a particular Quality-of-Service level agreement. A threshold completion time and/or a threshold variance of the completion time can be indicated by the selected pricing structure. Accordingly, the determined partition of the workflow can be a partition that has an expected completion time and/or a corresponding expected variance of the completion time that is at least under the threshold completion time and/or a threshold variance of the completion time specified in the Quality-of-Service level agreement associated with the pricing structure.

[0083] At 786, the method 780 can include allocating the workloads to each of the plurality of executors according to the determined partition. Allocating the workloads can include splitting the workflow into workloads determined by the determined partition value and disseminating the workloads to the executors for execution.

[0084] In the disclosure, reference is made to the accompanying drawings that form a part hereof, and in which is shown by way of illustration how a number of examples of the disclosure can be practiced. These examples are described in sufficient detail to enable those of ordinary skill in the art to practice the examples of this disclosure, and it is to be understood that other examples can be used and that process, electrical, and/or structural changes can be made without departing from the scope of the disclosure.

[0085] The figures herein follow a numbering convention in which the first digit corresponds to the drawing figure number and the remaining digits identify an element or component in the drawing. Elements shown in the various figures herein can be added, exchanged, and/or eliminated so as to provide a number of additional examples of the disclosure. In addition, the proportion and the relative scale of the elements provided in the figures are intended to illustrate the examples of the disclosure, and should not be taken in a limiting sense.

[0086] As used herein, "logic" is an alternative or additional processing resource to perform a particular action and/or function, etc., described herein, which includes hardware, e.g., various forms of transistor logic, application specific integrated circuits (ASICs), etc., as opposed to computer executable instructions, e.g., software firmware, etc., stored in memory and executable by a processor. Further, as used herein, "a" or "a number of" something can refer to one or more such things. For example, "a number of widgets" can refer to one or more widgets. Also, as used herein, "a plurality of something" can refer to more than one of such things.

[0087] The above specification, examples and data provide a description of the method and applications, and use of the system and method of the disclosure. Since many examples can be made without departing from the spirit and scope of the system and method of the disclosure, this specification merely sets forth some of the many possible configurations and implementations.

What is claimed is:

1. A method, comprising:
 - determining, using a machine, an expected completion time and a corresponding expected variance of the expected completion time of an execution of a portion of a workflow for each of a plurality of executors based on a corresponding execution speed of the executor and a property of the portion of the workflow;
 - determining, using the machine, a partition of the workflow into workloads to be allocated among the plurality of executors that achieves a workflow completion time under a threshold completion time; and
 - allocating the workloads to each of the plurality of executors according to the determined partition.
2. The method of claim 1, wherein the plurality of executors is a plurality of heterogeneous executors.
3. The method of claim 1, including determining the corresponding expected variance based on a fluctuation in an execution speed associated with each of the plurality of executors executing a respective portion of the workflow, wherein the plurality of executors are a plurality of parallel executors.
4. The method of claim 1, further comprising determining, using the machine, an updated completion time and an updated corresponding expected variance based on and evaluation of a length of time the workflow takes to complete a component associated with the machine.
5. A non-transitory machine-readable medium storing instructions executable by a processing resource to:
 - determine an expected completion time for a workflow and a corresponding expected variance of the completion time for the workflow across a plurality of partition values;

determine a partition value of the plurality of partition values that produces a shortest expected completion time and a smallest corresponding expected variance of the completion time; and

partition the workflow into workloads allocated to each of a first plurality of executors associated with the workflow according to the determined partition value.

6. The non-transitory machine-readable medium of claim 5, the instructions further executable to:

determine an execution speed and a corresponding variance of the execution speed of each of a second plurality of executors; and

determine a property of a workflow associated with the second plurality of executors.

7. The non-transitory machine-readable medium of claim 5, the instructions further executable to join together the completed workloads upon completion of execution by the first plurality of executors to form a completed workload

8. The non-transitory machine-readable medium of claim 5, wherein the first plurality of executors is a plurality of heterogeneous executors.

9. The non-transitory machine-readable medium of claim 6, wherein the first plurality of executors is a plurality of heterogeneous executors and the second plurality of executors is a plurality of homogeneous executors.

10. A system, comprising:

a first estimation engine to estimate a processing capability of each of a plurality of system components;

a workflow engine to partition a workflow associated with the plurality of system components into a plurality of workloads;

a component engine to run each of the plurality of workloads through each of the plurality of components;

an observation engine to evaluate how each of the plurality of components handles each of the plurality of workloads; and

a second estimation engine to update the processing capability estimates of each of the plurality of system components based on evaluation.

11. The system of claim 10, further comprising a receipt engine to: receive a different workflow; and

update the workflow partition based on the updated processing capability estimates.

13. The system of claim 10, the workflow engine to randomly partition the workflow.

14. The system of claim 10, the first estimation engine to estimate the processing capability of each of the plurality of system components using a Bayesian approach.

15. The system of claim 10, wherein the processing capability of each of the plurality of system components is a workload completion time of each of the plurality of workloads.

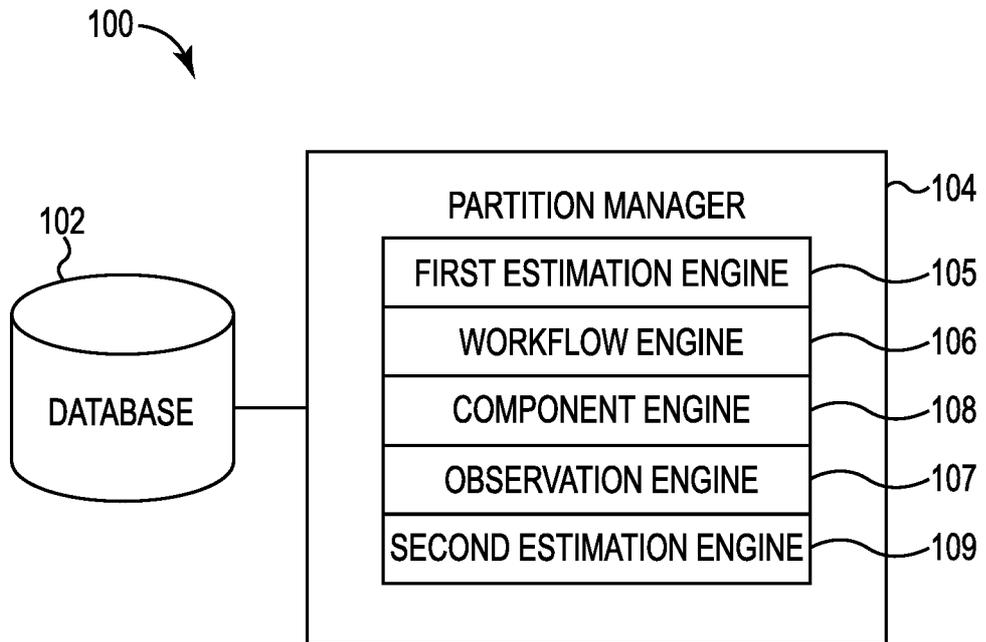


Fig. 1

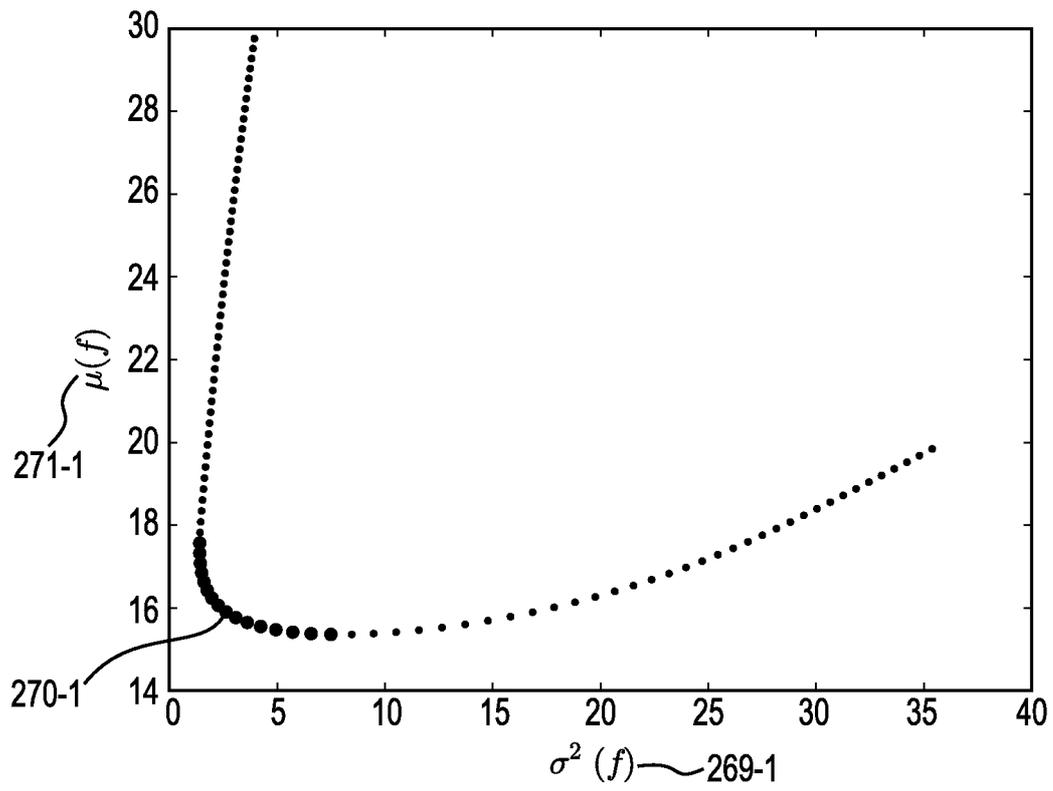


Fig. 2A

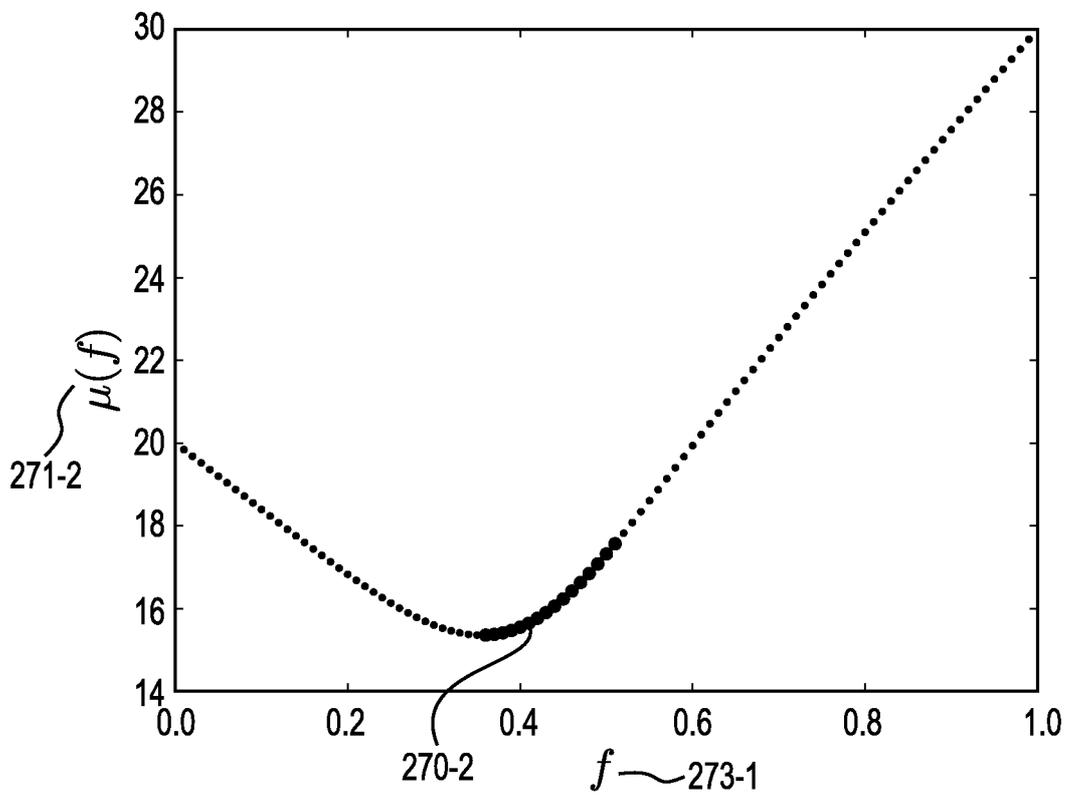


Fig. 2B

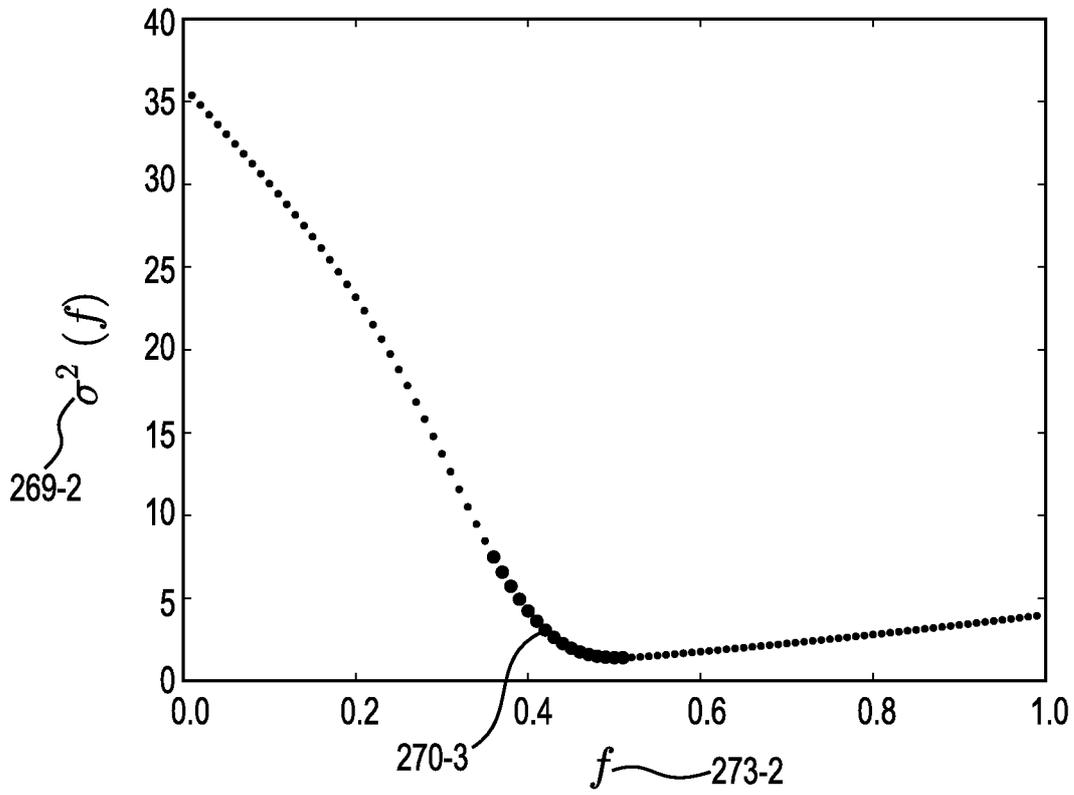


Fig. 2C

4/8

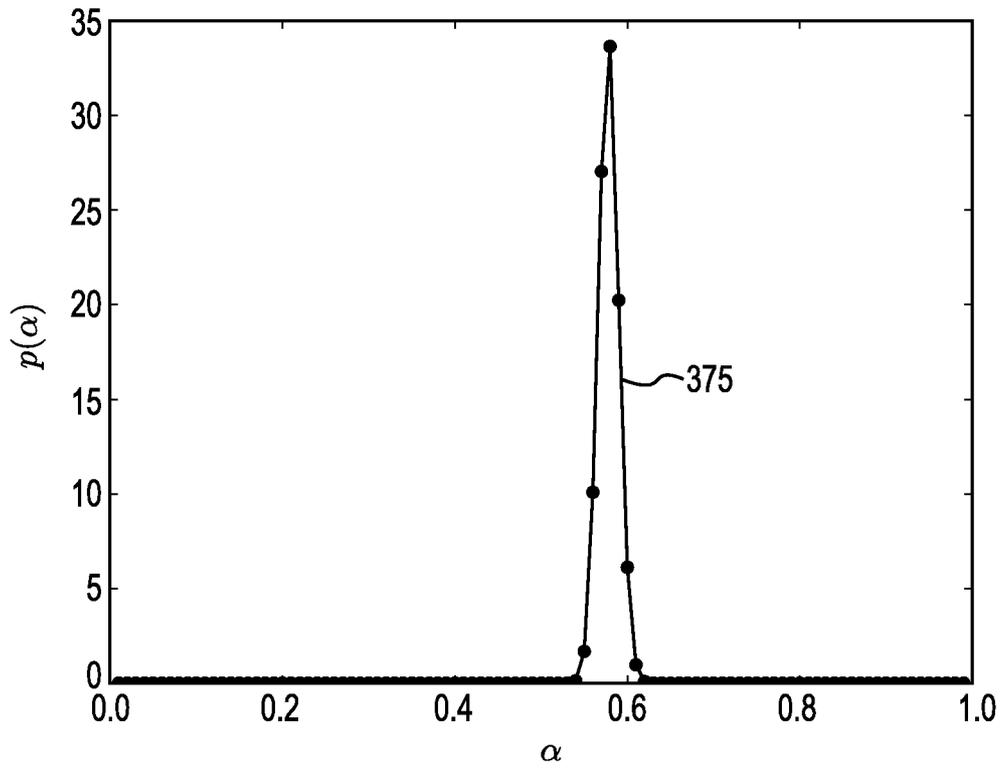


Fig. 3A

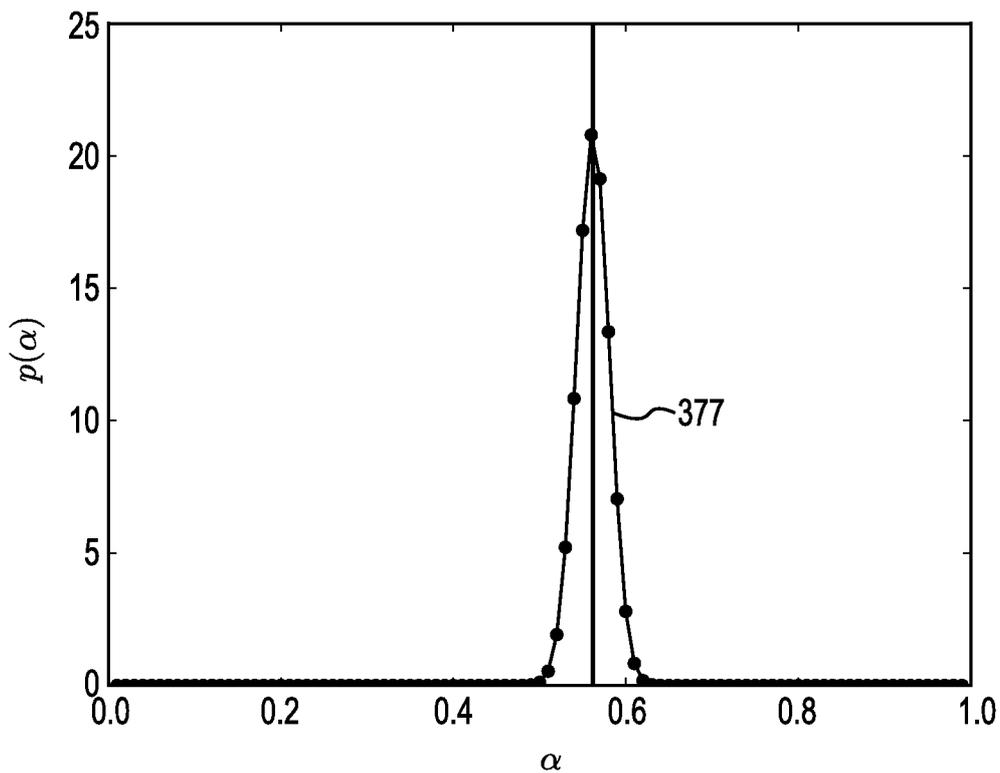


Fig. 3B

5/8

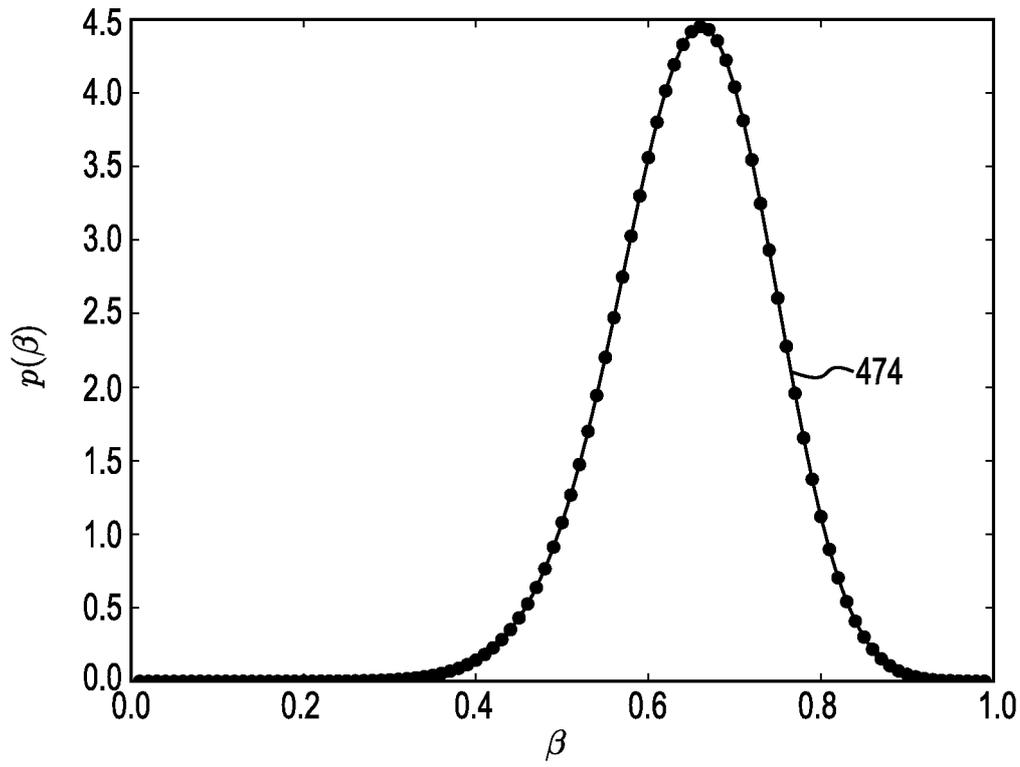


Fig. 4A

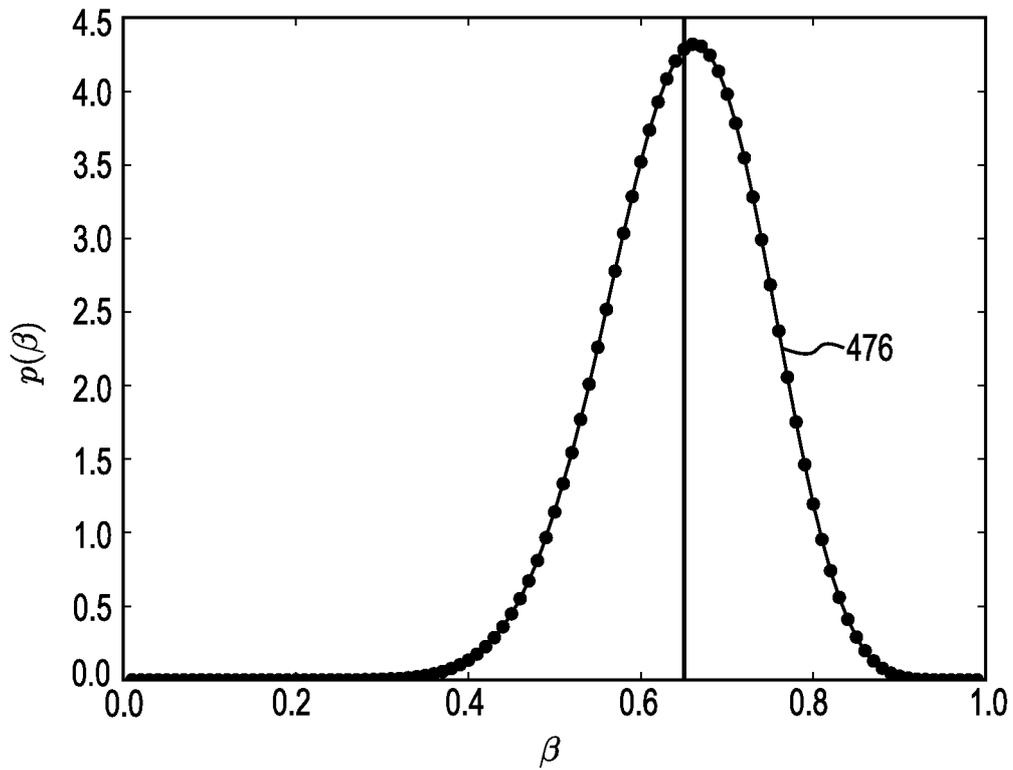


Fig. 4B

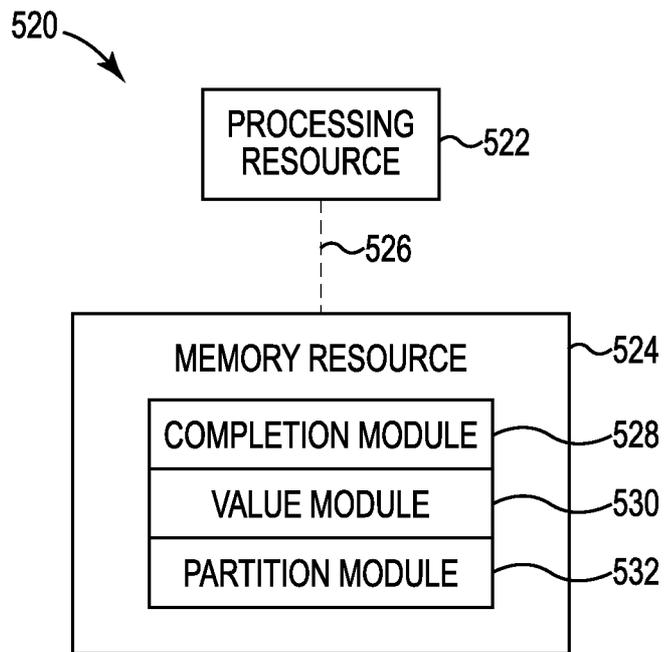
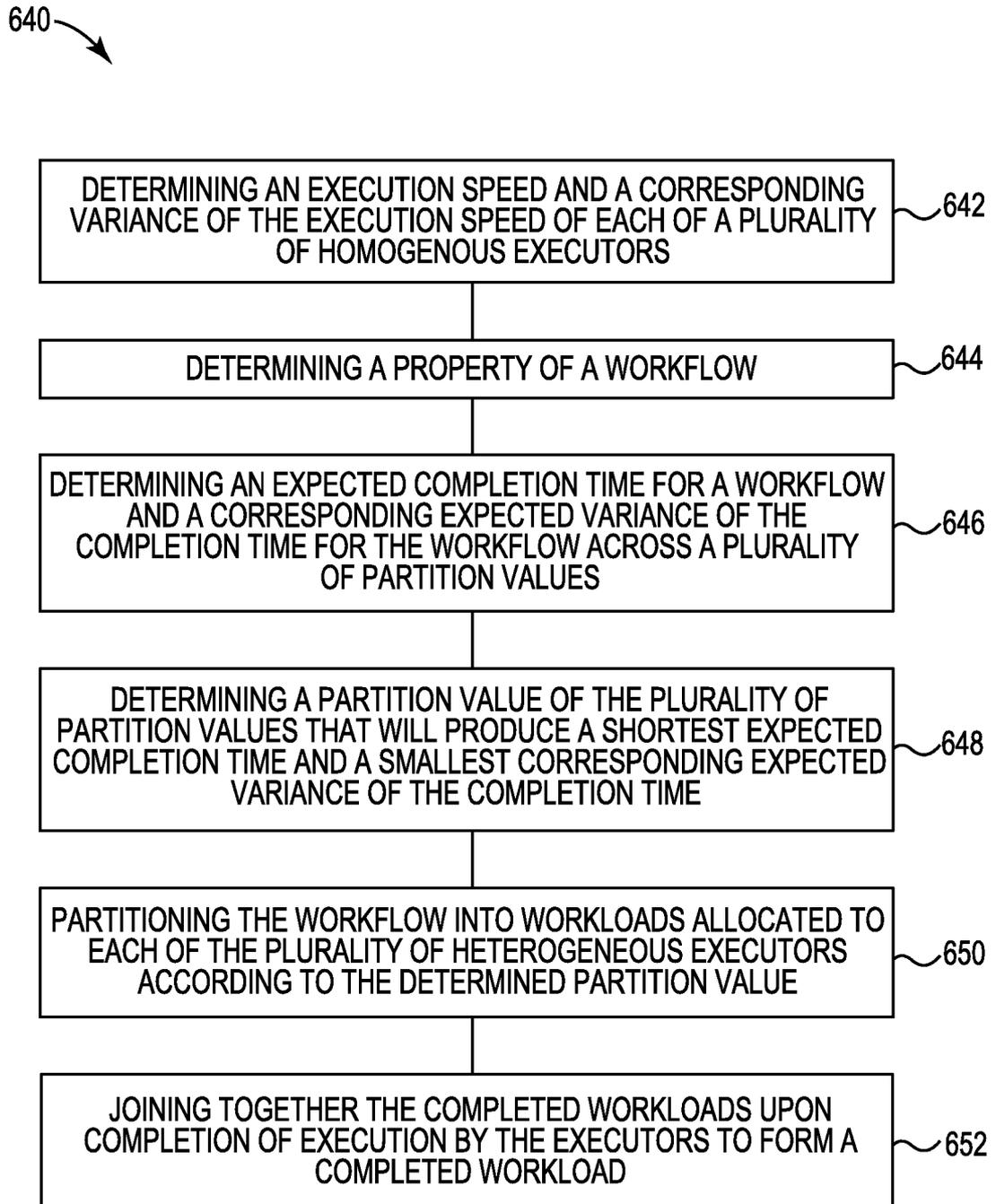
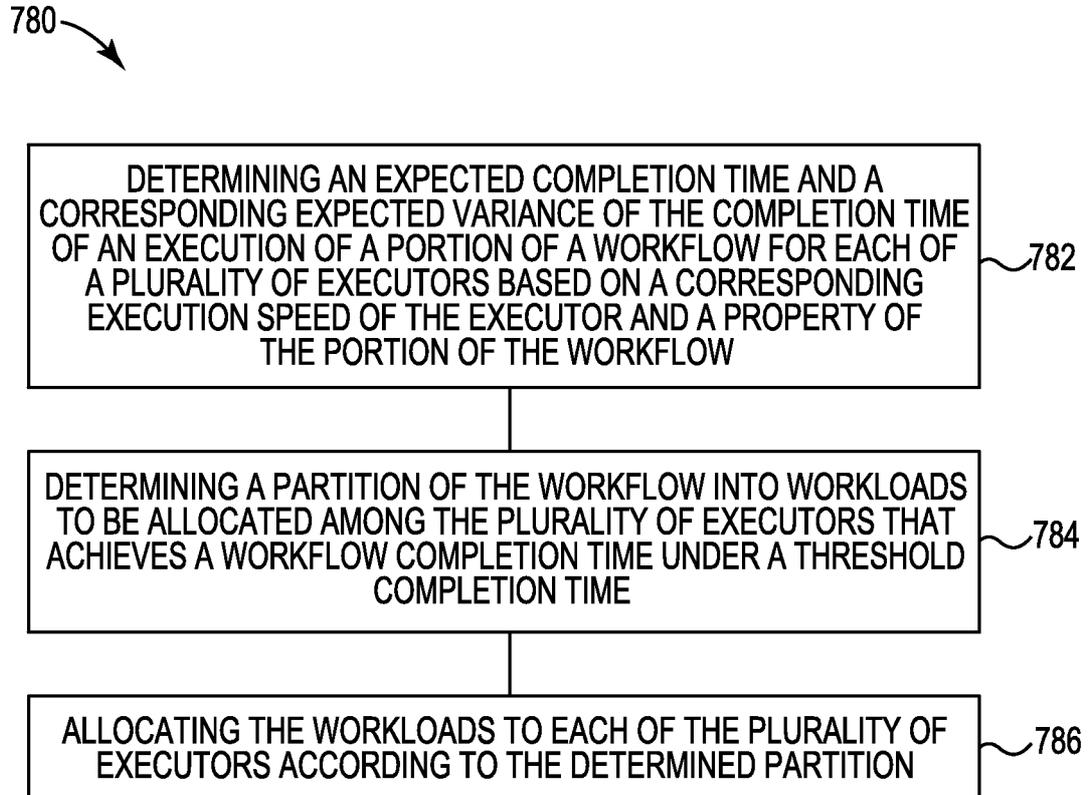


Fig. 5

7/8

**Fig. 6**

8/8

**Fig. 7**

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2015/057494**A. CLASSIFICATION OF SUBJECT MATTER****G06F 9/48(2006.01)i, G06F 9/46(2006.01)i**

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F 9/48; G06F 19/00; G06F 17/00; G06F 17/60; G06F 1/26; G06F 9/46

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Korean utility models and applications for utility models

Japanese utility models and applications for utility models

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

eKOMPASS(KIPO internal) & Keywords : expected completion time, expected variance, workflow partitioning, partition value, workload allocating

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category ^a	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2003-0153994 A1 (LI-JIE JIN et al.) 14 August 2003 See abstract; paragraphs [0053]-[0068]; claim V, and figures 4-6.	1-11,13-15
A	US 7035808 B1 (JON ALLAN FORD) 25 April 2006 See abstract; column 3, line 51 - column 4, line 50, column 11, line 42 - column 12 line 64; claim V, and figures 1, 4-5.	1-11,13-15
A	US 2014-0089694 A1 (APPLE INC.) 27 March 2014 See abstract; paragraphs [0034]-[0044]; claim V, and figures 1-3.	1-11,13-15
A	US 2004-0237087 A1 (NONG YE et al.) 25 November 2004 See abstract; paragraphs [0020]-[0026]; claim 11; and figure 3.	1-11,13-15
A	US 6601035 B1 (EUTHIMIOS PANAGOS et al.) 29 July 2003 See abstract; column 2, line 42 - column 3, line 11; claim V, and figures 2A-2C.	1-11,13-15

 Further documents are listed in the continuation of Box C. See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

22 July 2016 (22.07.2016)

Date of mailing of the international search report

22 July 2016 (22.07.2016)

Name and mailing address of the ISA/KR

International Application Division
Korean Intellectual Property Office
189 Cheongsa-ro, Seo-gu, Daejeon, 35208, Republic of Korea

Facsimile No. +82-42-481-8578

Authorized officer

CHIN, Sang Bum

Telephone No. +82-42-481-8398



INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2015/057494

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2003-0153994 AI	14/08/2003	US 7127716 B2	24/10/2006
US 7035808 BI	25/04/2006	CA 2322723 AI	20/04/2001
		EP 1094655 AI	25/04/2001
		JP 2001-155084 A	08/06/2001
		KR 10-2001-0051142 A	25/06/2001
US 2014-0089694 AI	27/03/2014	None	
US 2004-0237087 AI	25/11/2004	None	
US 6601035 BI	29/07/2003	None	