

Génération automatique de modèles calibrés. Une méthodologie complète

Thomas Begin, Bruno Baynat, Alexandre Brandwajn, Serge Fdida, Safia Kedad-Sidhoum, Francis Sourd

▶ **To cite this version:**

Thomas Begin, Bruno Baynat, Alexandre Brandwajn, Serge Fdida, Safia Kedad-Sidhoum, et al.. Génération automatique de modèles calibrés. Une méthodologie complète. CFIP 2006 - 12ème Colloque Francophone sur l'Ingénierie des Protocoles, Eric Fleury and Farouk Kamoun, Oct 2006, Tozeur, Tunisie. pp.1-12. inria-00111725

HAL Id: inria-00111725
https://inria.hal.science/inria-00111725

Submitted on 20 Nov 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Génération automatique de modèles calibrés

Une méthodologie complète

Thomas Begin* — **Bruno Baynat*** — **Alexandre Brandwajn**** — **Serge Fdida*** — **Safia Kedad*** — **Francis Sourd***

** Laboratoire Informatique de Paris 6*
Université Pierre et Marie Curie Laboratoire d'Informatique de Paris 6 - CNRS
8, rue du Capitaine Scott
75015 Paris

{thomas.begin, bruno.baynat, serge.fdida, safia.kedad-sidhoum, francis.sourd}@lip6.fr

*** University of California, Santa Cruz*

alexb@cse.ucsc.edu

RÉSUMÉ. *Nous présentons une méthodologie pour générer automatiquement des modèles à partir de mesures. Elle repose sur une approche descriptive et s'articule autour de trois notions. Premièrement les briques : elles constituent des modèles génériques présupposés dont seront extraits les modèles finaux. Deuxièmement le critère d'erreur : il permet d'apprécier la qualité des modèles. Enfin, le processus de calibration a pour but de trouver automatiquement le modèle d'une brique qui minimise le critère d'erreur (c'est-à-dire la meilleure combinaison de ses paramètres). Nous explorons ces notions en y apportant des solutions et notamment, en définissant une technique originale de calibrage des paramètres dont nous évaluerons les performances. Le document se termine par la présentation des modèles rendus par notre méthodologie sur des exemples réels. Ces modèles obtenus peuvent servir à la planification de capacité.*

ABSTRACT. *This work aims at defining and exploring a methodology that would allow the user to automatically generate models from a given set of measurement data. The proposed approach involves three major points. First, we define a set of generic models, referred to as "bricks". Second, we propose a criterion by which we evaluate the quality of the model in terms of how well it fits the measurement data and the objective of the study. The third point is the calibration process whose goal is to select the best combination of parameters for the "brick" considered. In this paper, we discuss these points, focusing in particular on an original tool for model calibration. We conclude by presenting examples of models obtained through our methodology from real measurement data. Such models may be of use in capacity planning issues.*

MOTS-CLÉS : *Modélisation, Performances, Approche descriptive, Calibration automatique, Files d'attente*

KEYWORDS: *Modeling, Performance, Descriptive approach, Automatic calibration, Queueing*

CFIP 2006 - POST PRINT

1. Introduction

Les modèles analytiques sont ordinairement utilisés lors de la conception de nouveaux systèmes, lors du dimensionnement de systèmes existants ou bien dans l'optique d'une *planification de capacité* sur un système existant. Cette dernière opération consiste à estimer le comportement d'un système en dehors de sa plage de fonctionnement habituel (par exemple pour estimer sa capacité à monter en charge). Pour toutes ces opérations le processus de modélisation fait en général appel à une approche constructive. Le modèle pensé vise à reproduire le fonctionnement interne du système. Ce processus se heurte parfois à l'exigence d'une connaissance fine du système ainsi qu'au besoin d'une expertise spécifique pour identifier les points-clés du système dans le comportement étudié. Ces difficultés inhérentes à l'approche constructive nous ont incité à rechercher une méthodologie pour modéliser les systèmes à comportement de type file d'attente en nous basant uniquement sur des observations extérieures. Il existe des techniques qui utilisent uniquement des mesures pour estimer les caractéristiques internes d'un système. [DOV 01] qui évalue la capacité d'un chemin Internet (en réalité de son goulot d'étranglement) en fait partie. D'autres techniques font appel à un a priori. [ALO 01] présuppose un *modèle d'inférence* de type file d'attente pour fournir des estimations sur la capacité d'un chemin Internet et sur l'intensité du trafic concurrent. Dans nos travaux le modèle d'inférence recherché n'est pas décidé au regard du système comme dans [ALO 01]. Il devra être déterminé et ses paramètres devront être calibrés. Le système devra exister pour fournir des mesures mais aucune connaissance sur son fonctionnement interne n'est requise. Cette approche est qualifiée de « *descriptive* » [SAL 03] en raison du rôle central qu'elle confère aux mesures.

Nous présentons une méthodologie systématique, rapide et automatique pour modéliser le comportement de systèmes pour lesquels on dispose de points de mesures. Nos travaux visent à démontrer la faisabilité de cette approche « boîte noire » qui représente une alternative simple à la modélisation par approche constructive notamment pour la planification de capacité. Le résultat de notre méthode permettra de prédire simplement et rapidement le comportement du système étudié en dehors des points de mesures.

On désigne par *client* l'entité du système à laquelle on s'intéresse. Sa définition est étroitement liée au comportement étudié du système. Par exemple, les clients peuvent correspondre à des paquets IP dans les réseaux informatiques ou bien à des requêtes dans un serveur. La Figure 1 illustre les paramètres de performances moyens classiquement étudiés : la *charge de travail* en entrée λ correspondant au nombre moyen de clients qui arrivent à l'entrée du système par unité de temps, le débit moyen de sortie $\bar{X}$ correspondant au nombre moyen de clients qui quittent le système par unité de temps (qui peut être différent de la charge de travail en entrée si le système est à perte [1]), la probabilité de rejet Pr des clients en entrée du système pour les systèmes à perte, le temps de séjour moyen $\bar{R}$ d'un client accepté dans le système et le nombre moyen de clients simultanément présents dans le système $\bar{Q}$. Les mesures à la base de notre méthodologie sont toutes issues d'un même système (en régime permament) et seront indicées par $_{mes}$. Elles constituent une série de n mesures mettant en correspondance plusieurs paramètres de performances moyens à des niveaux de charge

1. Des clients peuvent se voir refuser l'accès au système à leur arrivée.

CFIP 2006 - POST PRINT

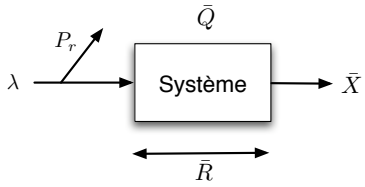


Figure 1. *Mesure d'un système*

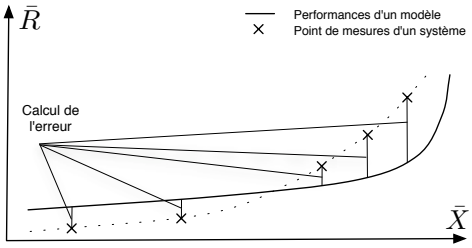


Figure 2. *Calcul de l'erreur*

différents. Les paramètres du système (par exemple le nombre de serveurs) doivent être identiques pour toutes les mesures ou bien varier selon des règles bien définies, reliées à la charge soumise.

La Section 2 présente les étapes et les outils qui composent notre méthodologie. La Section 3 décrit un instrument de calibrage automatique en insistant sur sa généricité et sur ses originalités. Les résultats obtenus par notre méthodologie sur des cas réels sont présentés et analysés dans la Section 4. La conclusion, en Section 5, évoque les directions envisagées pour les travaux futurs.

2. Méthodologie de modélisation automatique

Contrairement à [ALO 01], le modèle d'inférence retenu par notre méthodologie sera choisi parmi un ensemble de modèles génériques présupposés. Nous nommerons *brique* chacune de ces familles de modèles génériques et nous réserverons le terme *modèle* pour l'association d'une brique et d'un *calibrage* (i.e. une combinaison particulière des paramètres de la brique). Notre méthodologie s'articule autour de trois axes. Premièrement le critère d'erreur, il permet de juger de la qualité d'un modèle (en rapport aux mesures) et de comparer les modèles entre eux. Deuxièmement une méthode automatique et générique de calibrage d'une brique, elle a pour but de trouver rapidement pour chaque brique la combinaison de ses paramètres qui minimise le critère d'erreur. Enfin l'ensemble de briques, il constitue le réservoir initial de modèles génériques. Nous avons opté pour des briques de type file d'attente car notre méthodologie s'adresse à des systèmes dans lesquels les clients occasionnent des délais de service et d'attente.

Pour illustrer notre méthodologie nous supposons n couples de mesures ($\bar{X}_{mes,i}$, $\bar{R}_{mes,i}$) correspondant à n charges d'entrée différentes d'un système. Notons qu'à partir de la loi de Little : $\bar{Q} = \bar{R}\bar{X}$ ([BAY 00], [KLE 75]), il serait tout à fait équivalent de disposer de mesures du type ($\bar{X}_{mes,i}$, $\bar{Q}_{mes,i}$), ($\bar{R}_{mes,i}$, $\bar{Q}_{mes,i}$) ou bien ($\bar{X}_{mes,i}$, $\bar{Q}_{mes,i}$, $\bar{R}_{mes,i}$). Plus généralement, la démarche présentée peut être adaptée à tout autre ensemble de mesures.

2.1. *Critère d'erreur*

Il s'agit d'une fonction visant à apprécier la qualité d'un modèle, c'est-à-dire sa capacité à reproduire les performances mesurées du système. L'usage de ce critère permet de comparer avec la même objectivité les modèles intra-familles (deux calibrages différents d'une même brique) et inter-familles (deux briques différentes).

CFIP 2006 - POST PRINT

Une façon très simple d'estimer l'erreur consiste à calculer la somme cumulée des écarts entre les performances théoriques du modèle et celles mesurées sur le système. Soit $\bar{R}_{th,i}$ le temps moyen de séjour théorique du modèle obtenu à partir d'une charge d'entrée λ_i qui conduit à un débit de sortie théorique égal au débit de sortie mesuré [2] : $\bar{X}_{th,i} = \bar{X}_{mes,i}$. L'erreur sera la somme des écarts résiduels entre les $\bar{R}_{mes,i}$ et les $\bar{R}_{th,i}$ comme l'illustre la Figure 2. Nous nommerons f la fonction erreur :

$$f = \sum_{i=1}^{n} |\bar{R}_{th,i} - \bar{R}_{mes,i}| \qquad [1]$$

Lorsque les mesures le permettent ou le préconisent, l'analyste peut décider de redéfinir son critère d'erreur. L'analyste peut choisir un critère d'erreur combinant écarts absolus et écarts relatifs. Un poids α (resp. $1-\alpha$) compris entre 0 et 1 permettra de contrôler la contribution du critère aboslu (resp. relatif) dans le calcul du critère d'erreur moyen. Il est par ailleurs possible d'associer à chaque point de mesure un degré de confiance par l'intermédiaire d'une pondération w_i sur la mesure i. En tenant compte de ces aménagements, la définition de la fonction erreur devient :

$$f = \alpha \sum_{i=1}^{n} w_i |\bar{R}_{th,i} - \bar{R}_{mes,i}| + (1-\alpha) \sum_{i=1}^{n} w_i \left| \frac{\bar{R}_{th,i} - \bar{R}_{mes,i}}{\bar{R}_{mes,i}} \right| \qquad [2]$$

Notons que si les mesures sont accompagnées d'intervalles de confiance, ceux-ci peuvent être également pris en compte dans la fonction erreur. Il est par exemple possible de réduire le poids associé à une mesure dans le calcul de l'erreur si, pour cette mesure, le $\bar{R}_{th}$ se situe à l'intérieur de l'intervalle de confiance du $\bar{R}_{mes}$ associé.

Dans la suite de ce document nous adoptons la fonction f de [2] qui comprend une combinaison des écarts relatifs et absolus (avec α égal à 0.5) et dont tous les w_i sont choisis égaux de façon à ne privilégier aucun point en particulier.

2.2. *Recherche du meilleur modèle parmi toutes les briques*

2.2.1. *Recherche modulaire*

Pour chaque brique nous recherchons le calibrage (i.e. la combinaison des paramètres) qui minimise le critère d'erreur f. Une fois le meilleur modèle pour chaque brique connu, le critère d'erreur permet également d'établir un classement inter-brique qui présente par ordre de préférence les modèles retenus par notre approche. Si du meilleur calibrage de deux briques résulte deux modèles au niveau d'erreur très proche, nous choisirons le candidat le plus simple (nombre de paramètres le moins élevé).

2.2.2. *Exemple de brique : la M/M/C*

Nous présentons ici, à titre d'exemple, une des briques les plus simples. La M/M/C est une file d'attente de capacité illimitée, comportant C serveurs en parallèle, chacun ayant un temps de service exponentiellement distribué, et soumise à un processus d'arrivée des clients poissonien de taux λ (cf Figure 3). Elle se caractérise par deux paramètres : un paramètre discret C égal au nombre de serveurs et un paramètre continu μ égal au taux moyen de service de chaque serveur. La condition de stabilité : $\lambda < C\mu$ doit être satisfaite pour analyser son régime permament [BAY 00], [KLE 75].

2. Si la brique est un modèle ouvert sans perte alors $\lambda_i = \bar{X}_{mes,i}$. Dans le cas d'un modèle à perte en entrée, la charge d'entrée devra être inférée sur le débit de sortie (par des techniques de dichotomie par exemple).

CFIP 2006 - POST PRINT

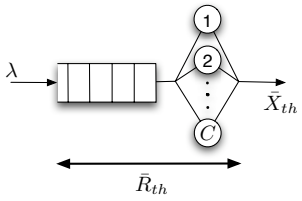


Figure 3. *Une file d'attente M/M/C*

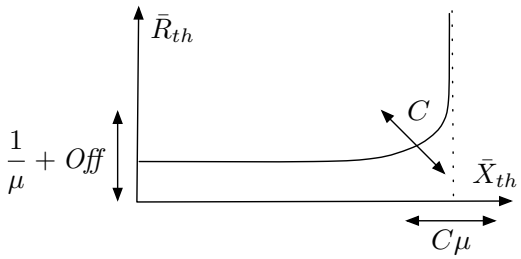


Figure 4. *La courbe de performances d'une M/M/C*

De l'analyse stationnaire de la file, on en déduit l'expression des paramètres de performances moyens analytiques (notées $\bar{X}_{th}$ et $\bar{R}_{th}$), ainsi que les propriétés remarquables suivantes :

– $\bar{X}_{th} = \lambda$: puisqu'il n'y a pas de perte ;

– $\bar{R}_{th} > \frac{1}{\mu}$: puisque le temps de séjour dans la file est au moins égal au temps moyen de service ;

– $\bar{X}_{th} < C\mu$: puisque la file est stable.

La *courbe de performances*, terme que nous associons dans ce document à la représentation de $\bar{R}_{th}$ en fonction de $\bar{X}_{th}$, d'une M/M/C présente une allure monotone et bimodale avec deux régimes bien différenciés. Trois grandeurs caractéristiques la contrôlent comme le souligne la Figure 4 : $\frac{1}{\mu}$ pour l'asymptote horizontale résultant du service quasi-immédiat des clients à faible charge, $C\mu$ pour l'éloignement de l'asymptote verticale marquant la saturation et C qui décide du degré de courbure de la zone intermédiaire. Or dans sa forme initiale, le modèle M/M/C ne dispose que de deux paramètres. Nous compensons ce manque par l'ajout d'un degré de liberté supplémentaire matérialisé par un paramètre : *l'Offset temporel*. C'est une constante qui s'ajoute au temps de séjour des clients dans tous les modèles mais sans intervenir dans les calculs d'attente. On peut assimiler sa présence à un temps de traitement incompressible et indépendant de la charge (un temps de transfert par exemple). Son intégration à toutes nos briques est très simple, elle revient à placer une M/M/∞ ([BAY 00], [KLE 75]) en série derrière le modèle principal. Notons qu'il s'agit d'un premier pas vers la composition de briques simples.

2.2.3. *Espace de recherche borné*

Des contraintes sur les paramètres des briques réduisent l'espace de recherche de la combinaison optimale. Par exemple pour une M/M/C, l'obligation de respecter la condition de stabilité $\lambda < C\mu$ et le fait que pour chaque point de mesure i on ait, $\lambda_i = \bar{X}_{th,i} = \bar{X}_{mes,i}$, impose que les paramètres C et μ du modèle soient tels que : $C\mu > \max_{i\in[1;n]} \bar{X}_{mes,i}$. De même, l'inégalité $\bar{R}_{th} > \frac{1}{\mu} + Off$ est toujours vérifiée pour une M/M/C (cf. Figure 4) et pour que les mesures soient cohérentes avec le modèle, il faudra que : $\frac{1}{\mu} + Off < \min_{i\in[1;n]} \bar{R}_{mes,i}$. Les *bornes*, générées par ces contraintes, peuvent être relâchées du fait de l'incertitude inhérente à une mesure en s'assurant que le modèle reste applicable. Dans ce document nous relâchons les bornes de 10%.

CFIP 2006 - POST PRINT

2.2.4. *Recherche automatique du calibrage d'une brique*

Une première méthode de calibrage automatique commune à toutes les briques est une approche exhaustive et *systématique*. Très simple à mettre en œuvre, cette technique s'appuie sur une discrétisation des paramètres continus par un pas dont la taille est difficile à estimer a priori. Autre handicap, elle nécessite de déterminer une borne inférieure et une borne supérieure pour chacun des paramètres du modèle ce qui n'est pas toujours possible. Enfin sa complexité exponentielle la rend inappropriée dès l'instant que la brique possède plus de 3 ou 4 paramètres (se rapporter au Tableau 1). Il devient alors nécessaire de recourir à une autre technique. Nous optons pour une méthode de descente par voisinage que nous présentons en détails dans la Section 3.

2.3. *Présentation succincte de quelques briques possibles*

Nous présentons ici uniquement des briques simples car les jeux de mesures qui ont accompagné nos recherches présentaient des comportements assez caractéristiques des systèmes d'attente (voir Figure 7 pour les jeux de mesures). Cette sélection constitue un aperçu restreint des briques que nous avons définies mais elle suffit à montrer la faisabilité de l'approche. L'approche modulaire et générique de notre méthodologie permet d'ajouter facilement d'autres briques tout en conservant le caractère automatique de la méthode. Toutefois il est clair qu'aucun ensemble de briques, aussi vaste soit-il, ne pourra satisfaire à tous les jeux de mesures.

– La brique M/M/∞ est une M/M/C avec un nombre infini de serveurs chacun de taux μ. Ainsi le temps d'attente des clients est toujours nul et le temps de séjour moyen des clients toujours égal à $\frac{1}{\mu}$ quelle que soit la charge.

– La M/M/C/K correspond à une M/M/C pour laquelle la capacité serait rendue finie (de taille K) ce qui en fait un modèle à perte en entrée. L'égalité entre λ et $\bar{X}_{th}$ est rompue mais étant donné la croissance stricte de $\bar{X}_{th}$ avec λ, il est simple d'inférer par dichotomie la charge en entrée λ pour un débit de sortie donné $\bar{X}_{mes}$.

– Nous avons également défini des modèles fermés, M/M/C//N, pour lesquels le nombre de clients est fixe. L'extérieur est modélisé par une file supplémentaire à N sources. Pour ces modèles le taux d'arrivée des clients n'est plus poissonien.

– Nous avons aussi des modèles de type M/G/1. Ici la loi de service du serveur suit une distribution quelconque. La connaissance de ses deux premiers moments (ou de manière identique, sa moyenne et son coefficient de variation : cv) suffit à évaluer les paramètres de performance moyens de cette file (formule de Pollaczek Kinchin [BAY 00], [KLE 75]).

Toutes ces briques (sauf la M/M/∞ qui n'a pas d'intérêt en soi mais modélise l'Offset) présentent des courbes de performances ($\bar{R}_{th}$ fonction de $\bar{X}_{th}$) strictement croissantes. C'est un comportement qui est en adéquation avec nos jeux de mesures et plus généralement avec ceux fréquemment rencontrés dans des systèmes d'attente (dû à la rivalité des clients pour accéder aux ressources). Plus spécifiquement la courbe de performances d'une M/M/∞ se réduit à une droite horizontale à la hauteur $\frac{1}{\mu}$. Celle d'une M/M/C/K ou d'un modèle M/M/C//N aura une allure proche de celle d'une M/M/C exceptée la présence d'un point d'accumulation qui se substitue à l'asymptote verticale de saturation. Cette différence résulte de l'existence d'un temps de séjour moyen maximum pour ces modèles (égal à $\frac{K}{C\mu}$ pour une M/M/C/K et à $\frac{N}{C\mu}$ pour un M/M/C//N) à mesure que $\bar{X}_{th}$ s'approche de la capacité maximale de traitement. Quant aux modèles M/G/1, leur courbe de performances dépend beaucoup de la valeur

CFIP 2006 - POST PRINT

du cv de la loi de service. Ils présentent un comportement bimodal pour des faibles valeurs de cv tandis qu'ils ont tendance à saturer plus graduellement à mesure que la valeur de cv augmente (le ralentissement sur $\bar{R}_{th}$ intervient dès les petites charges en entrée). Toutes ces informations engendrent des bornes sur les paramètres de chacune des briques similaires à celles présentées pour la M/M/C.

3. Technique d'optimisation *ad hoc* pour le calibrage des paramètres

3.1. *Formulation du problème et état de l'art*

Notre méthodologie impose la résolution de plusieurs problèmes de régression non-linéaires (pour chacune des briques il faut découvrir la combinaison optimale de ses paramètres). Pour cela nous avons opté pour la solution suivante. Soit une brique quelconque ayant m paramètres. L'ensemble des combinaisons possibles de ces m paramètres, soumises aux contraintes exprimées par les bornes, constitue l'espace de recherche des solutions. A chacun des points de cet espace correspond une valeur de la fonction erreur f. Un point de cet espace sera d'autant meilleur que cette valeur est petite. Nous considérons à présent la surface (ou pour être précis l'hypersurface de dimension m dans l'espace de dimension $m+1$) qui réunit l'ensemble des points de l'espace des solutions (cf Figure 5, les lignes de niveaux de f font apparaître plus clairement l'allure de la surface). Notre objectif consiste à trouver efficacement sur cette surface un point proche du plus petit des minima. Les coordonnées de ce point traduiront les valeurs des m paramètres de la brique qui permettent de la calibrer au mieux. Soit $x = (\mathit{param}_1, \mathit{param}_2, \ldots \mathit{param}_m)$ un vecteur des m paramètres inclus dans l'espace de recherche des solutions et f la *fonction objectif* à minimiser représentant l'erreur associée au modèle (un exemple de f, celui que nous utilisons dans ce document, est donné éq. [2]). Pour pouvoir bénéficier des travaux effectués en optimisation numérique, nous avons rendu continus tous les paramètres des briques car ces techniques le nécessitent. Nous avons défini des *modèles intermédiaires* dont tous les paramètres sont continus et qui coincident avec les modèles classiques quand les valeurs des paramètres sont entières[3]. Calibrer au mieux une brique revient à résoudre un problème d'*optimisation numérique non linéaire avec contraintes de bornes* :

$$\min_{x \in \mathbb{R}^m} f(x)$$

Les algorithmes d'optimisation classiques utilisent les dérivées de f. Hélas ses dérivées analytiques sont difficilement calculables (voire impossibles à calculer) et leur spécificité à chaque brique empêcherait l'ajout aisé de nouvelles briques dans notre méthodologie. Les techniques d'approximation des dérivées, comme [KAP 00], sont écartées à cause du coût de calcul important de f (il nécessite l'évaluation des performances d'un modèle à plusieurs niveaux de charge) et, a fortiori, f peut ne pas être différentiable pour certains modèles.

3. Certains paramètres, comme C pour une M/M/C, peuvent être indifféremment entiers ou continus dans l'expression des formules de performances (calcul de $\bar{X}_{th}$, calcul de $\bar{R}_{th}$). Pour d'autres, comme pour le paramètre K d'une M/M/C/K, la chaîne de Markov doit être étendue afin de supporter la continuité du paramètre et de générer un modèle aux performances intermédiaires. En dernier recours il est toujours possible d'estimer la valeur d'un paramètre continu à partir des valeurs des paramètres entiers les plus proches par interpolation.

CFIP 2006 - POST PRINT

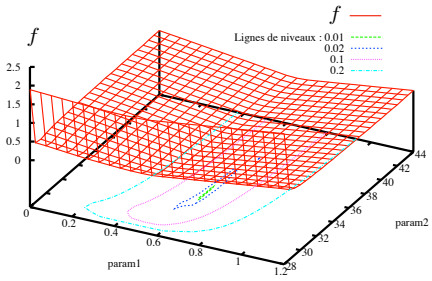


Figure 5. *Exemple de f pour une brique à deux paramètres*

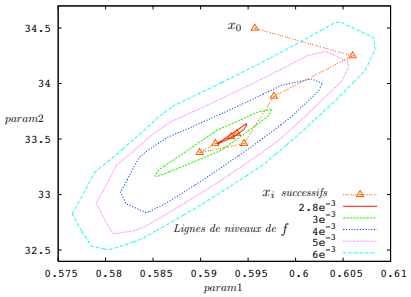


Figure 6. *Exemple de descente (DFO)*

Il reste la piste des techniques d'optimisation sans dérivée (dites *DFO*, « derivate-free-optimization ») [CON 97]. On y trouve les techniques qui procèdent à la construction d'un modèle sans passer par le calcul (exact ou approché) d'un gradient ou d'une dérivée. Ces techniques tentent d'approcher la fonction sur une région dite de confiance par un modèle lisse et régulier et d'utiliser ce modèle pour optimiser la fonction sous-jacente. [POW 74] démontre que dans certains cas ces techniques s'appuyant sur *un modèle de substitution* s'avèrent plus efficaces que les techniques de gradients ou de dérivées. Nous avons opté pour ce type de technique en choisissant comme modèle de substitution la *quadrique*. Notons que f n'étant en général pas convexe, aucune méthode polynômiale ne peut garantir systématiquement la convergence vers le minimum global (sauf si P=NP).

Au lieu de calquer le jeu de paramètres utilisé par notre technique de calibrage sur le jeu de paramètres de la brique, on opte pour un jeu qui contient les grandeurs caractéristiques des performances associées à la brique. Ainsi, pour une M/M/C on a choisi le jeu de paramètres $(C\mu, \frac{1}{\mu} + \mathit{Off}, C)$ au lieu de (C, μ, Off) car il reflète les corrélations existantes entre les paramètres et l'allure de la courbe de performances. Ce changement de variables permet de rendre notre méthode, DFO, plus robuste et plus rapide.

3.2. *Principe de résolution*

Nous nous limitons à une description succincte de l'algorithme. La technique que nous proposons opère par améliorations successives, on parle aussi de *descente*. Elle démarre d'un point (vecteur de paramètres) quelconque x_0 de l'espace des solutions[4]. A chaque itération on recherche un point plus bas que le point itéré courant x_i qui devient le nouveau point courant x_{i+1} jusqu'à satisfaction d'une condition d'arrêt[5]. Pour inférer x_{i+1} on construit un modèle simple qui vise à reproduire l'allure de la fonction objectif f. Le modèle mathématique adopté est une quadrique dont le caractère polynomial d'ordre 2 rend la manipulation rapide et simple.

On définit le *voisinage* de x_i comme la zone de confiance au sein de laquelle l'approximation quadratique est supposée valide. A chaque paramètre du modèle est

4. x_0 est fixé en faisant appel à des heuristiques.
5. La Figure 6 présente un exemple de convergence par améliorations successives vers le minimum de f pour une brique à 2 paramètres.

CFIP 2006 - POST PRINT

associé un pas p_j qui détermine le rayon du voisinage dans cette direction : le voisinage forme donc un ellipsoïde. On choisit p points, y_1, ...y_p dans le voisinage x_i, que l'on nomme les *points d'interpolation*, et qui servent à construire le modèle de substitution, c'est-à-dire à déterminer les valeurs des paramètres de la quadrique. Les y_j sont choisis à la surface de l'ellipsoïde et on en choisit aléatoirement un nombre p qu'on complète avec x_i de façon à disposer du nombre de points nécessaire et suffisant pour caractériser de manière unique une quadrique en dimension m (c'est-à-dire $p+1 = 2m+\frac{m(m-1)}{2}+1$ points). Notons que si un point y_j est tel que : $f(y_j) < f(x_i)$, alors $x_{i+1} = y_j$.

Une rapide résolution d'un système d'équations permet de trouver l'expression analytique de la quadrique unique passant par tous les y_j et par x_i. Ensuite on infère les coordonnées du minimum (unique) de cette quadrique, $\tilde{x}_i$. Si $f(\tilde{x}_i) < f(x_i)$, le minimum préconisé par le modèle de substitution est meilleur que le point courant et donc $x_{i+1} = \tilde{x}_i$. Dans le cas contraire, on intensifie la recherche en réduisant la taille des pas p_j. A mesure que la recherche avance, la taille de la zone de confiance dont sont extraits les points y_j doit être réduite pour affiner localement la précision du modèle quadratique et atteindre des points très proches du minimum vers lequel on converge. Ce processus d'intensification est régi par des heuristiques. On réduit le pas du paramètre qui provoque la plus grande augmentation de f que l'on suppose être celui de la direction la plus abrupte.

Les conditions d'arrêt de la méthode de descente retenues sont :

– Le point préconisé par la quadrique coïncide presque avec le point itéré courant x_i, c'est-à-dire : $f(\tilde{x}_i) - f(x_i) < \epsilon$ et $\|x_i - \tilde{x}_i\| < \epsilon$ où ϵ est une petite constante positive arbitrairement choisie.

– Les pas associés aux paramètres sont petits : $\max_{j\in[1;p]} p_j < \epsilon$ et les points y_j et x_i ont des valeurs de f quasi-identiques : $\forall j \in [1;p], ||f(y_j) - f(x_i)|| < \epsilon$.

4. Résultats numériques

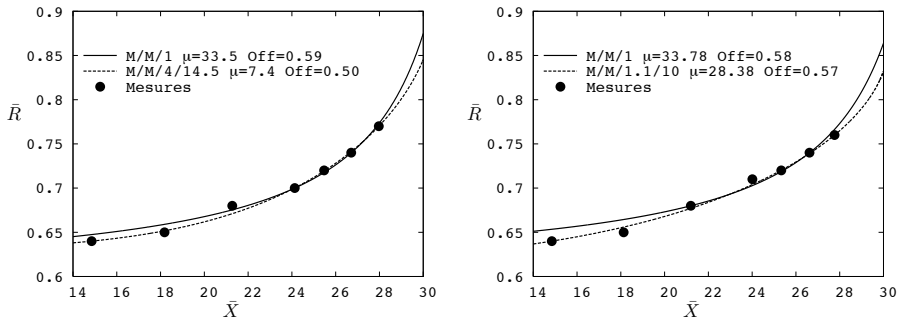
4.1. *Présentation des jeux de mesures*

Dans cette section nous mettons à l'épreuve notre méthodologie sur des cas réels. Ces scénarios correspondent à des jeux de mesures observés sur des contrôleurs disques. Les requêtes d'entrées/sorties jouent le rôle des clients. Sur le plan qualitatif ces jeux de mesures présentent des comportements classiques de systèmes avec attente. Plus le débit mesuré augmente, plus la charge en entrée est supposée grande (on a égalité seulement dans les systèmes sans perte), et plus l'attente d'un client pour être servi est longue (aux erreurs de mesures près). C'est un comportement symptomatique de *saturation* d'un système [6].

Un examen plus précis des jeux de mesures montre que leurs allures comportent des différences. Les jeux de mesures 1 et 2, Figures 7(a) et 7(b), ont un premier stade où les $\bar{R}_{mes,i}$ sont maintenus quasi-constants lorsque les $\bar{X}_{mes,i}$ augmentent (puis la saturation sur les $\bar{R}_{mes,i}$ démarre). A l'inverse sur les jeux 3 et 4, Figures 7(c) et 7(d), le phénomène de saturation semble se déclencher plus tôt : dès les premiers $\bar{X}_{mes,i}$, on note un ralentissement des $\bar{R}_{th,i}$, qui sera amplifié sur les $\bar{X}_{mes,i}$ suivants. Ces diffé-

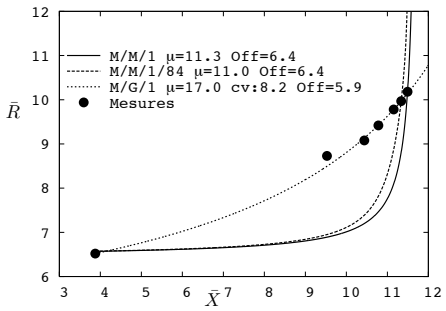
6. Globalement seuls les systèmes qui mettent à disposition de nouvelles ressources en réaction à une charge importante dérogent à cette règle de fonctionnement.

CFIP 2006 - POST PRINT

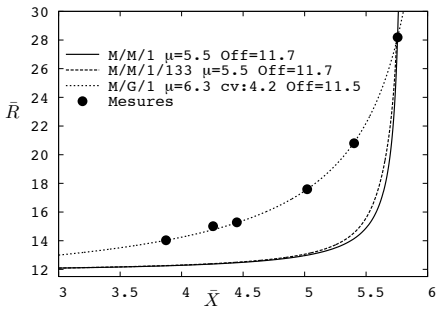


(a) Jeu de mesures 1

(b) Jeu de mesures 2



(c) Jeu de mesures 3



(d) Jeu de mesures 4

Figure 7. *Modèles obtenus sur les jeux de mesures 1 à 4*

rences qualitatives sont automatiquement prises en compte par notre méthodologie à travers le type de la brique finale retenue.

4.2. *Evaluation de la technique de calibrage* ad hoc

Nous avons évalué les performances de notre technique de calibrage (DFO) présentée en Section 3 sur des combinaisons possibles de briques et de jeux de mesures. Nous avons également confronté ses performances avec celles d'une approche systématique et celles d'un algorithme d'optimisation numérique basé essentiellement sur les gradients : Solvopt [7] [KAP 00]. Les facteurs de performances que nous avons mesurés sont la *vitesse de convergence* (estimée par le nombre d'appels moyen à f : $\#f$ et le temps CPU moyen en seconde[8]) et la *robustesse* (pourcentage de réalisations qui convergent vers un point x_i tel que $f(x_i) < 1.15 \min f$ indiqué par Rob.). Ces valeurs seront calculées à partir de milliers de réalisations en faisant varier x_0, le point d'amorce, dans tout l'espace des solutions. Le paramétrage des méthodes d'optimisation est choisi au mieux grâce à des résultats expérimentaux et reste toujours le même. Pour l'approche systématique il s'agit essentiellement de fixer la taille des pas et pour notre technique DFO d'initialiser les p_j et ϵ. La robustesse de l'approche systématique ne sera pas évaluée car nous avons choisi empiriquement le pas de discrétisation le plus grand qui assure la découverte d'un bon calibrage.

7. http ://bedvgm.kfunigraz.ac.at :8001/alex/solvopt/index.html
8. programmé en C et exécuté sur un serveur DualCore AMD Opteron 275

CFIP 2006 - POST PRINT

Jeu	Brique	DFO			Solvopt			Systématique	
		CPU	$\#f$	Rob.	CPU	$\#f$	Rob.	CPU	$\#f$
1	M/M/C	$4.7e^{-2}$	3513	100%	$1.3e^{-2}$	1300	14%	2.03	$4.62e^{5}$
1	M/M/C/K	1.55	6424	69%	1.57	2794	13%	1176	$1.85e^{7}$
2	M/M/C	$4.5e^{-2}$	3377	100%	$1.2e^{-2}$	2149	26%	2.1	$4.68e^{5}$
2	M/M/C/K	1.52	6418	85%	1.51	2842	25%	1197	$1.88e^{7}$
3	M/G/1	$5.1e^{-2}$	3687	85%	$1.05e^{-2}$	1180	27%	523	$7.12e^{8}$
4	M/G/1	$4.9e^{-2}$	3471	100%	$2.12e^{-2}$	3217	33%	4591	$6.49e^{9}$

Tableau 1. *Performances comparées des trois techniques de calibrage sur des combinaisons de briques et de jeux de mesures*

Les résultats du Tableau 1 font apparaître notre méthode comme le meilleur compromis vitesse - robustesse, rivalisant avec Solvopt pour la vitesse (bien que toujours inférieur) et la dominant nettement pour la robustesse. Nous constatons également que les bonnes performances de notre méthode sont maintenues lorsque le nombre de paramètres des briques augmente (les résultats obtenus pour la brique M/G/1 sont très bons). Notons que pour comparer les capacités de robustesse des techniques d'optimisation, nous les avons volontairement testées dans des conditions difficiles en prenant les points d'amorce x_0 dans tout l'espace de recherche. Pour une utilisation courante, l'appel à des heuristiques sur les briques permet de fixer un « bon » x_0, situé à proximité de l'optimum recherché. Le taux de réussite des optimisations est alors très largement augmenté (très proche de 100% pour notre méthode DFO). En outre des techniques simples comme le « multi départ » [MAY 88] permettent de mieux contrôler les chances de succès de la méthode.

4.3. *Modèles retenus par notre méthodologie*

Pour les jeux de mesures 1 et 2, le calibrage des briques M/M/C et M/M/C/K rend compte de très bons résultats. Les courbes de performances des modèles obtenus pour ces briques se situent très près de l'ensemble des points comme le montrent les Figures 7(a) et 7(b). Notons que d'autres briques, plus compliquées, n'apportent que très peu d'améliorations à cette approximation. En revanche les jeux de mesures 3 et 4 mettent en échec les briques M/M/C, M/M/C/K et M/M/C//N (l'exemple des briques M/M/C et M/M/C/K en témoigne). Le calibrage le plus favorable de ces briques engendrent des courbes de performances très éloignées de certains points de mesures. La logique du système n'est pas respectée. Par conséquent on fait appel à des briques plus sophistiquées telles la M/G/1 qui permet plus de flexibilité sur la montée en saturation. La M/G/1 permet d'obtenir un modèle calibré très proche des mesures du jeu 3 et du jeu 4 comme l'illustrent les Figures 7(c) et 7(d).

L'exploitation de ces modèles pour la planification de capacité passe par la lecture des courbes de performances. Le modèle lauréat de notre méthodologie (dont l'existence n'est pas garantie pour chaque jeu de mesures) servira à prédire le comportement du système étudié en dehors des points de mesures. L'analyste dispose très rapidement et très simplement d'une prédiction quant aux performances du système mesuré et ce quelque soit le niveau de charge qui l'intéresse.

CFIP 2006 - POST PRINT

5. Conclusions et travaux futurs

Les résultats de la Section 4 montrent la faisabilité de l'approche descriptive pour certains usages et nous encouragent à poursuivre le développement de notre méthodologie. L'approche descriptive, au coeur de nos travaux, ne s'oppose pas à l'approche constructive. Rien n'empêche d'intégrer des informations supplémentaires sur le système dans ce type d'approche. Par exemple, on peut changer un paramètre d'une brique en une constante si l'on souhaite respecter une valeur réelle connue du système (comme le nombre de serveurs d'un système) ou bien exclure les briques de modèles à perte si le système est sans perte. Notre technique de calibrage par un procédé d'optimisation numérique utilise des procédés de DFO classiques et des heuristiques propres au contexte d'utilisation. Elle constitue une approche originale pour calibrer atomatiquement des modèles analytiques. Sa capacité à inférer efficacement le calibrage optimal de briques à 4 ou 5 paramètres en fait une pièce maîtresse de la méthodologie. Son aspect très générique permet de l'exploiter pour toutes les briques existantes et à venir.

Les travaux futurs viseront à multiplier le nombre de cas d'étude (probants ou pas) dans des domaines variés aux spécificités singulières (serveurs web, trafic réseau, etc). Ils porteront aussi sur le sens à donner aux paramètres des modèles d'inférence retenus par notre méthodologie. Les résultats peuvent-ils apporter une perception intrusive sur le système ? D'autre part nous poursuivrons la définition de nouvelles briques. Mais cette recherche doit se faire très méticuleusement. Une brique simple suffisamment proche des mesures suggère que le comportement étudié du système s'apparente à un modèle de file d'attente simple. Pour une brique compliquée légèrement meilleure il devient difficile de savoir si c'est le calibrage ad hoc aux mesures ou bien la structure de la brique qui a permis d'approcher les mesures.

6. Bibliographie

[ALO 01] ALOUF S., NAIN P., TOWSLEY D. F., « Inferring Network Characteristics via Moment-Based Estimators », *INFOCOM*, 2001, p. 1045-1054.

[BAY 00] BAYNAT B., *Théorie des files d'attente, des chaînes de Markov aux réseaux à forme produit*, Hermes Science Publications, Paris, France, 2000.

[CON 97] CONN A. R., SCHEINBERG K., TOINT P. L., « Recent progress in unconstrained nonlinear optimization without derivatives », *Math. Program.*, vol. 79, n° 1-3, 1997, p. 397–414, Springer-Verlag New York, Inc.

[DOV 01] DOVROLIS C., RAMANATHAN P., MOORE D., « What Do Packet Dispersion Techniques Measure ? », *INFOCOM*, 2001, p. 905-914.

[KAP 00] KAPPEL F., KUNTSEVICH A. V., « An Implementation of Shor's r-Algorithm », *Comput. Optim. Appl.*, vol. 15, n° 2, 2000, p. 193–205, Kluwer Academic Publishers.

[KLE 75] KLEINROCK L., *Queueing Systems Volume 1 : Theory*, Wiley, 1975.

[MAY 88] MAYNE D., MEEWELLA C., « A non-clustering multistart algorithm for global optimization », , 1988, p. 334–345, Springer-Verlag.

[POW 74] POWELL M. J. D., « Unconstrained Minimization Algorithms Without Computation of Derivatives », *Bollettino della Unione Matematica Italiana*, vol. 9, 1974, p. 60–69.

[SAL 03] SALAMATIAN K., FDIDA S., « A framework for interpreting measurement over Internet », *Proceedings of the ACM SIGCOMM workshop on Models, methods and tools for reproducible network research*, NY, USA, 2003, ACM Press, p. 87–94.

CFIP 2006 - POST PRINT