

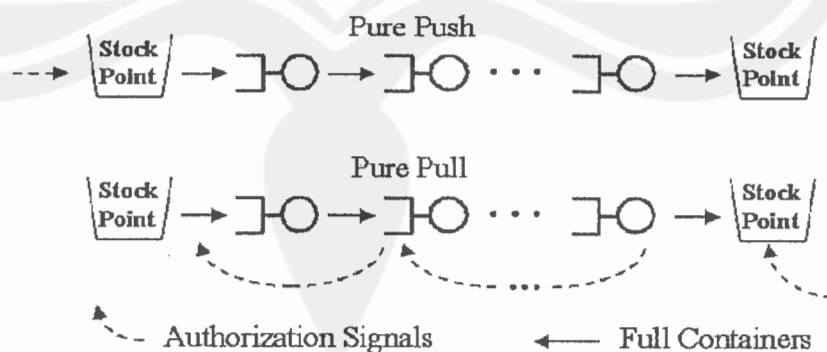
BAB 3

LANDASAN TEORI

3.1. Sistem *Push* dan *Pull*

Sistem *push* me-release pekerjaan yang telah dijadwalkan. Sistem *push* ditentukan oleh jadwal produksi induk, karena release berhubungan dengan pesanan ataupun hasil peramalan. *Push* system dikendalikan oleh informasi *upstream* dan tidak dapat dipisahkan dari system produksi *make-to-order*. Dalam sistem *Pull*, status sistem menentukan apakah suatu pekerjaan dapat diijinkan untuk *release* atau tidak. Intinya, status system ditentukan oleh persediaan, dimana system *pull* dikendalikan oleh informasi *downstream* dan tidak dapat dipisahkan oleh system produksi *make-to-stock*. Sistem *pull* *buffer* bertindak sebagai persediaan kosong untuk me-release suatu pekerjaan (<http://www.factoryphysics.com/Principle/Station/Asynch/Open/InfinOpen/IBOLPushPull.htm>, diakses tanggal 11 Agustus 2007).

Gambar 3.1. *Push* versus *Pull* System



Secara sederhana, system push didasarkan pada order konsumen (permintaan) dan peramalan. Berdasarkan permintaan, perencanaan material dan kapasitas dapat dibuat. Lantai produksi dikendalikan oleh permintaan dan pesanan-pesanan. Aliran material dan informasi menuju arah yang sama. Ini menunjukkan sistem produksi *make-to-stoks*. MRP adalah sistem *push*. Suatu sistem *pull* kontras didasarkan pada pesanan. Lantai produksi dikendalikan oleh kebutuhan aktual dan system produksi tepat waktu (Just In Time). Sistem *pull* menggunakan permintaan yang nyata (Customer Order Received), dimana pada sistem *push* didasarkan pada hasil peramalan dan membuat pesanan-pesanan yang diharapkan (<http://blogs.ittoolbox.com/erp/planning/archives/is-mrp-a-push-system-or-pull-system-10571>, diakses pada tanggal 7 November 2007).

3.1.1. Push Vs Pull System

Venkatesh (1996) memandang perbedaan antara system *push* dan sistem *pull* sekedar sebagai paradigma operasional. Pada sistem *push*, sebuah mesin melakukan proses produksi tanpa harus menunggu permintaan dari mesin yang akan melakukan proses berikutnya. Sebaliknya pada sistem *pull*, sebuah mesin melakukan proses produksi hanya jika ada permintaan dari mesin yang akan melakukan proses selanjutnya (Wibisono, 2004).

3.2. Inventori

3.2.1. Pendahuluan

Inventori atau persediaan dapat didefinisikan sebagai bahan yang disimpan dalam gudang untuk kemudian dapat digunakan atau dijual. Persediaan dapat berupa bahan baku untuk keperluan proses, barang-barang yang masih dalam pengolahan dan barang jadi yang disimpan untuk penjualan. Persediaan adalah hal yang pokok sebagai fungsi yang tepat dari suatu usaha pengolahan/pembuatan (Biegel, 1992)

Sepanjang mengadakan pembuatan atau pengolahan akan selalu memakan waktu, dimana persediaan akan selalu terdiri dari barang-barang proses pengolahan. Dalam beberapa industri bahan-bahan harus diolah dalam lot atau dalam bentuk tumpukan-tumpukan. Dalam industri lainnya aliran bahan harus disiapkan untuk menghasilkan produk sekaligus dalam beberapa tahapan kelengkapan. Dalam cara pengolahan lainnya persediaan adalah diperlukan sekali, baik dari pertimbangan-pertimbangan ekonomis maupun untuk memproses atau menjadwalkan bahan dalam bentuk lot (Biegel, 1992).

Sifat produk, sifat dari permintaan konsumen dan sifat proses pembuatan menentukan suatu pertimbangan untuk perluasan yang diperlukan dalam persediaan barang jadi. Kebanyakan pabrik pengolahan mengadakan persediaan-persediaan lainnya yang tidak langsung dijadikan sebagai produk. Persediaan seperti ini terdapat dalam bentuk perkakas, alat-alat pendukung, peralatan dan lain-lainnya (Biegel, 1992).

Persediaan yang terdiri dari berbagai bentuk, penanganan persediaannya menimbulkan banyak masalah. Tujuan pengendalian dan perencanaan persediaan dan produksi adalah untuk mencari penyelesaian terhadap masalah-masalah tersebut. Dalam cara ini, pengendalian produksi berkaitan dengan perencanaan operasi produksi, pemindahan dan penyimpanan barang-barang. Hubungan langsung antara tingkat persediaan, jadwal produksi, dan permintaan konsumen, perencanaan dan pengendalian persediaan harus digabungkan dengan peramalan permintaan, perencanaan produksi dan pengendalian produksi (Biegel, 1992).

3.2.2. Fungsi Persediaan

Persediaan membantu untuk memisahkan pemasok, produsen, dan konsumen. Persediaan juga mengizinkan pengadaan bahan baku dalam ukuran lot ekonomis dan juga pengolahan bahan baku ini menjadi barang jadi dalam jumlah yang ekonomis. Persediaan bahan baku memisahkan pemasok bahan baku dari para pemakai bahan baku ini. Persediaan barang jadi memisahkan konsumen dari barang jadi tersebut (Biegel, 1992).

3.2.3. Tipe Persediaan

Menurut Biegel (1992) antara pemasok dan produsen, persediaan diadakan untuk:

- a. Pemindahan bahan.
- b. Pemisahan pemasok dan produsen.

c. Pembelian dan pengolahan barang-barang dalam jumlah ekonomis.

Persediaan barang jadi secara umum dibuat karena tiga alasan menurut Biegel (1992), yaitu:

- a. Untuk memungkinkan membuat barang dalam jumlah ekonomis.
- b. Untuk menyediakan permintaan atau penjualan di masa yang akan datang (perkiraan-persediaan).
- c. Untuk menyiapkan suatu penyangga dalam menghadapi gejolak permintaan nyata dari permintaan yang diramalkan (fluktuasi atau persediaan keamanan).

3.3. Instruksi-Instruksi Umum untuk simulasi *menggunakan Spreadsheets*

Instruksi umum ini menguraikan bagaimana cara menciptakan angka-angka acak untuk digunakan di simulasi, bagaimana cara menyusun *spreadsheet*, dan bagaimana cara memproses *output* dari simulasi. Fungsi dibahas di sini adalah: RAND, TRUNC, IF, AND, OR, and FREQUENCY (<http://math.colgate.edu/math102/dlantz/excel/generalsim.html> diakses pada tanggal 12 November 2007).

3.3.1. *Random Numbers*

Ada sedikitnya 3 cara untuk menciptakan angka-angka random dengan *Ms.Excel*, yaitu:

- a. Fungsi RAND() adalah fungsi yang dapat mengacak bilangan dengan nilai antara nol (0) sampai dengan satu (1). Kita dapat membuat bilangan acak di dalam sel manapun berubah setiap melakukan apapun di dalam

spreadsheet dengan menggunakan dengan fungsi `RAND()`. Hal ini disebabkan *spreadsheet* dapat "recalculating" atau menghitung ulang secara otomatis setiap kita melakukan sesuatu.

Bagaimana jika kita menginginkan bilangan acak yang bukan antara nol dan satu? Jika kita ingin bilangan acak berada pada nol (0) dan lima (5), maka kita dapat mengalikan fungsi `RAND()` dengan lima (5), yaitu `=5*RAND()`. Jika kita ingin bilangan acak berada pada tiga (3) dan delapan (8), gunakan `=3+5*RAND()`. Jika kita hanya menginginkan INTEGERS, kita harus membulatkan sistem desimal. Fungsi yang digunakan adalah `TRUNC()`. Fungsi `TRUNC()` menghilangkan semua decimal dari bilangan. Maka, `TRUNC(2*RAND())` dimulai nol setengah waktu, dan yang lainnya setengah dari waktu (Fungsi tadi sama hasilnya dengan fungsi `IF(RAND() $<$ 0.5,0,1]`).

Jika ingin mengganti dengan bilangan acak 1, 2, atau 3, gunakan fungsi `=1+TRUNC(3*RAND())`.

- b. Fungsi `RANDBETWEEN(lo,hi)` mengacak bilangan bulat dari angka yang paling rendah ke yang paling tinggi. (sebagai contoh, `RANDBETWEEN(1,5)` memulai bilangan bulat antara 1 dan 5). Seperti menggunakan fungsi `RAND()` akan menghitung ulang setiap kita melakukan apapun di dalam *spreadsheet*.
- c. Di dalam *Data Analysis Toolpak* terdapat fungsi yang disebut "Random Number Generation". Kita dapat menetapkan banyak sel yang diinginkan, distribusi yang

diperlukan, dan lain-lain dalam menentukan bilangan acak.

3.3.2. Logic Operators

Kita dapat menginterpretasikan jawaban-jawaban yang didapatkan dari simulasi yang menggunakan *logic operators*. Sebagai contoh, untuk menampilkan jika sel A3 dan B3 adalah sama, maka gunakan `=IF(A3=B3,1,0)`. Argumentasi pertama dari IF() adalah *expression* untuk mengevaluasi TRUE atau FALSE. Argumentasi kedua adalah bilangan yang artinya TRUE dan ketiga bilangan jika FALSE. Fungsi ini tidak harus berupa bilangan, kita juga dapat membuat seperti contoh berikut ini : `=IF(A3=B3,IF(A3=1,1,A1),RAND())` yang memberikan nilai satu (1) jika memasukkan nilai di dalam A3 dan B3 keduanya satu (1), akan memberikan nilai di dalam A1 jika A3 dan B3 memiliki nilai yang sama tetapi bukan satu (1), dan akan memberikan bilangan acak jika nilai di dalam A3 dan B3 tidak sama.

Terkadang kita menggunakan fungsi AND() dan OR() untuk membuat logika dengan fungsi IF(). Sebagai contoh, `=IF(AND(A3=B3,C3>2000),1,0)`. Fungsi AND() menjadi TRUE jika kedua *expression* TRUE. Fungsi OR() menjadi TRUE jika salah satu ataupun kedua *expressions* adalah TRUE.

3.3.3. Counting and Frequency

Jika kita ingin mengetahui berapa banyak sel yang mempunyai bilangan yang lebih besar atau kurang dari beberapa bilangan penggalan, kita dapat menggunakan

fungsi FREQUENCY(). Sebagai contoh, jika kita ingin mengetahui berapa banyak simulasi berakhir kurang dari -100 dan lebih besar dari nol dan lebih besar dari 100, fungsi FREQUENCY() yang digunakan.

Untuk menggunakan fungsi FREQUENCY(), kita harus mempunyai barisan data sel dan barisan penggalan yang merupakan batasan untuk sebuah "bins" yang ingin dikelompokkan di dalam. Daftar bilangan penggalan disebut dengan "bin array". Data array dan bin array harus dimasukkan ke suatu tempat. Sebagai contoh di atas bin array ada tiga sel dengan nilai-nilai -100,0,100 berturut-turut. Pastikan bahwa hanya terdapat satu bin sel-sel di dalam "bin array" karena bilangan-bilangannya adalah bilangan penggalan. Bin yang berlebih merupakan bilangan yang nilainya lebih dari bilangan penggalan. Dalam contoh yaitu bilangan yang nilainya lebih dari 100.

Perintahnya adalah =FREQUENCY(A101:G101,A102:C102) dimana A101:G101 adalah data array dan A102:C102 adalah bin array. Tetapi jangan menggunakan "function wizard" atau "Insert-->Function" dalam item menu. Untuk beberapa alasan, fungsi-fungsi tersebut tidak bekerja dengan benar.

Keluaran (output) dari fungsi FREQUENCY() menunjukkan banyaknya sel nilai bin sebelumnya dengan yang sekarang. Sebagai contoh, dengan bin array -100,0,100, output sel pertama berisi nilai bilangan dalam sel-sel yang nilainya kurang dari atau sama dengan -100; sel berikutnya berisi bilangan dengan nilai antara -100 and 0; berikutnya antara 0 and 100; sel terakhir berisi

banyaknya sel di dalam data array yang mempunyai nilai lebih besar dari penggalan yang terakhir, yaitu 100.

